

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



ThermInfo 2.0 - Estruturação e concretização de um sistema de informação para propriedades químicas.

Roni Wilson Salomão dos Reis

Projecto

Mestrado em Tecnologias de Informação aplicadas às
Ciências Biológicas e Médicas

2012

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



ThermInfo 2.0 - Estruturação e concretização de um sistema de informação para propriedades químicas.

Roni Wilson Salomão dos Reis

Projecto

Trabalho orientado pelo Prof. Doutor André Osório Falcão

Mestrado em Tecnologias de Informação aplicadas às
Ciências Biológicas e Médicas

2012

Resumo

O Grupo de Energética Molecular do Centro de Química e Bioquímica da Faculdade de Ciências da Universidade de Lisboa compilou, da literatura, um conjunto de dados termoquímicos experimentais de diversos compostos químicos. Estes dados serviram de base à implementação de um sistema de informação para coligir e apresentar propriedades termoquímicas, que foi denominado *ThermInfo*, e encontra-se disponível em <http://www.therminfo.com/>.

Recentemente o Grupo de Energética Molecular compilou mais dados de compostos químicos orgânicos, sendo que estes dados possuem novas propriedades para além das termoquímicas. Ficando comprometida a integração desses novos dados no sistema *ThermInfo*, estruturou-se a proposta deste trabalho: a estruturação de uma base de dados para propriedades químicas de compostos orgânicos, e a sua concretização no sistema *ThermInfo*, passando assim a *ThermInfo* 2.0, e por fim a reestruturação da arquitectura do sistema, separando o acesso aos dados da visualização dos mesmos. Este projecto integra quatro fases de desenvolvimento interdependentes e complementares: percepção (análise do problema e de requisitos), concepção (modelação da base de dados), implementação e avaliação.

A base de dados foi baseada numa especificação de dados relacional para descrever propriedades estruturais e químicas de compostos orgânicos e aplicada ao *ThermInfo* 2.0. Presentemente, a base de dados contém cerca de 30998 valores de propriedades, divididos por 20 propriedades químicas, para cerca de 11290 compostos orgânicos únicos e não redundantes.

Foi adoptado o modelo MVC (*Model-View-Controller*) para a nova arquitectura do sistema, que foi implementado através de um *framework* PHP (CodeIgniter). Os resultados da avaliação da base de dados e do sistema indicaram que estes são eficientes, em termos de tempo de execução e utilização de memória do servidor.

O desenvolvimento deste trabalho sugere algumas direcções futuras que irão ampliar as capacidades do sistema *ThermInfo* 2.0, nomeadamente a expansão do conjunto de dados integrando outras classes de compostos químicos, outros tipos de pesquisas na base de dados (InChI), e a construção de um *Web Service*.

Palavras-chave: Químio-informática, Base de Dados, Sistema de Informação, Propriedades Químicas, MVC.

Abstract

The Molecular Energy Group from the Chemistry and Biochemistry Center, Faculty of Science, University of Lisbon gathered, from literature, a data set of experimental thermochemical from several chemical compounds. These data were the basis for the implementation of an information system to collect and present structural and thermochemical properties of organic compounds, which has been named *ThermInfo*, and is available at <http://www.therminfo.com/>.

Recently the Molecular Energy Group has gathered more data of organic compounds, and these data have new properties in addition to the thermochemical. With problems to integrating these new data on the *ThermInfo*, the proposal of this work was structured: structuring a database for chemical properties of organic compounds, and its realization in the *ThermInfo* system, thus updated to *ThermInfo* 2.0, at last restructuring the system architecture, separating the data access from visualization. This project includes four interrelated and complementary phases of development: perception (problem and requirements analysis), design (database modeling), implementation and evaluation.

The database was based upon a relational data specification to describe structural and chemical properties of organic compounds and applied to *ThermInfo* 2.0. Currently, the database contains about 30998 property values, divided by 20 chemical properties, for about 11290 unique and non-redundant organic compounds.

It was adopted the MVC model (Model-View-Controller) for the new system architecture, which was implemented through a PHP framework (CodeIgniter). The evaluation results of the database and the system show that they are efficient, in terms of runtime and memory usage.

The development of this work suggests some future directions that will expand the capabilities of the *ThermInfo* 2.0 system, including the expansion of the data set with the integration of other classes of chemical compounds, other types of searches on the database (InChI), and implementation of a Web Service.

Keywords: Chemoinformatics, Database, Information System, Chemical Properties, MVC.

Agradecimentos

Gostava de aproveitar para expressar a minha gratidão ao conjunto de pessoas que contribuíram directa, ou indirectamente, para a realização do meu Mestrado e deste projecto:

- ao meu Professor e Orientador, Doutor André Osório Falcão, pela ajuda e empenho na realização deste projecto.
- um especial agradecimento a Ana Teixeira, pela paciência, ajuda e tempo dispensado na realização desde projecto.
- ao Grupo de Energética Molecular do Centro de Química e Bioquímica pela participação neste projecto.
- á minha namorada, Josilene Brito, por ter passado para segundo plano durante grande parte do projecto e por não me ter deixado desistir.
- á minha mãe Maria Das Dores Lucas, pelas restrições impostas a si mesma a fim de me auxiliar nesta missão.
- e um grande e especial agradecimento aos meus avós, Caetano Filipe de Sousa e Maria Isabel Filipe de Sousa, pelo apoio, amor e carinho ao longo da minha vida e pelo esforço dedicado para que eu pudesse chegar aqui.

Lisboa, 30 de Setembro de 2012

Conteúdo

LISTA DE FIGURAS.....	III
LISTA DE TABELAS	V
LISTA DE ABREVIATURAS E SIGLAS.....	VI
INTRODUÇÃO	1
1.1 MOTIVAÇÃO	1
1.2 OBJECTIVOS	2
1.3 METODOLOGIA	2
1.4 CRONOGRAMA.....	4
1.5 ORGANIZAÇÃO DO DOCUMENTO.....	4
CONCEITOS.....	6
2.1 THERMINFO	6
2.2 REPRESENTAÇÕES COMPUTACIONAIS DE ESTRUTURAS MOLECULARES.....	10
2.2.1 SMILES.....	11
2.2.2 SMARTS.....	12
2.2.3 InChI.....	12
2.2.4 MDL Molfile.....	13
2.2.5 Fingerprints.....	13
METODOLOGIA DE DESENVOLVIMENTO	15
3.1 ANÁLISE DO PROBLEMA	15
3.2 ANÁLISE DE REQUISITOS	15
3.2.1 Utilizadores.....	15
3.2.2 Funcionalidades	16
3.3 BASE DE DADOS	17
3.4 ARQUITECTURA DO SISTEMA	27
3.4.1 Model-view-controller	27
3.4.2 CodeIgniter	28
3.4.3 ThermInfo 2.0.....	30
3.5 INTERFACE.....	31
3.5.1 Front-end	31
3.5.2 Back Office.....	35
3.6 WEB SERVICE	36
RESULTADOS.....	39
4.1 BASE DE DADOS	39
4.2 SISTEMA	43
4.3 ANÁLISE	45
CONCLUSÕES	48
BIBLIOGRAFIA	50
APÊNDICES.....	52
A1 – MODELO COMPLETO DA BASE DE DADOS DO SISTEMA <i>THERMINFO 2.0</i>	52
A2 – CÓDIGO SQL PARA IMPLEMENTAÇÃO DA BASE DE DADOS EM MYSQL.....	54

Lista de Figuras

Figura 1 - Metodologia adoptada para o desenvolvimento do sistema.	3
Figura 2 - Página inicial do <i>ThermInfo</i>	6
Figura 3 - Interface da pesquisa simples.	7
Figura 4 - Interface da pesquisa por estrutura.	8
Figura 5 - Interface da pesquisa por propriedades termoquímicas.....	9
Figura 6 - Exemplo de uma lista de resultados de uma pesquisa.	9
Figura 7 - Interface da previsão de propriedades termoquímicas.....	10
Figura 8 - Exemplo do conteúdo de um <i>molfile</i> do composto benzeno.	13
Figura 9 - Visão global das funcionalidades da base de dados em termos de actores e dependências entre <i>Use-cases</i>	17
Figura 10 - Conjunto dos novos dados organizados em folhas de cálculo.....	17
Figura 11 - Diagrama de classes UML da base de dados.....	24
Figura 12 - Um diagrama exemplificando a relação entre o <i>Model</i> , <i>View</i> e <i>Controller</i> . 28	
Figura 13 - Fluxograma do CodeIgniter.	29
Figura 14 - Esquema da arquitectura do sistema <i>ThermInfo</i> 2.0.....	30
Figura 15 - Página inicial da interface do <i>ThermInfo</i> 2.0.....	31
Figura 16 - Formulário de ‘Pesquisa Simples’.	32
Figura 17 - Formulário de ‘Pesquisa por Estrutura’.	32
Figura 18 - Formulário de ‘Pesquisa Avançada’.	33
Figura 19 - Parte da listagem de resultados obtidos de uma pesquisa exemplo.	33
Figura 20 - Página com a ficha de um composto.	34
Figura 21 - Formulário para previsão de propriedades.....	34
Figura 22 - Parte da página com o resultado de uma previsão de propriedades.....	35
Figura 23 - Formulário que permite efectuar <i>login</i> no sistema.	35
Figura 24 - Página do <i>back office</i>	36
Figura 25 - Inserção de compostos no <i>back office</i>	36
Figura 26 - URI de acesso ao Web Service do <i>ThermInfo</i> 2.0.....	37
Figura 27 - Representação gráfica das estatísticas da base de dados (Registos).	39
Figura 28 - Representação gráfica das estatísticas das propriedades da base de dados (Registos).....	40
Figura 29 - Representação gráfica das estatísticas da base de dados (Carga).	43

Figura 30 - Representação gráfica das estatísticas do sistema (Execução).	44
Figura 31 - Representação gráfica das estatísticas do sistema (Memória).	44
Figura 32 - Representação gráfica das estatísticas do sistema (Carga).	45
Figura 33 - Modelo da base de dados do <i>ThermInfo</i> 2.0.	53

Lista de Tabelas

Tabela 1 - Cronograma das actividades do projecto (2011/2012).....	4
Tabela 2 - Tempo, em segundos, utilizado pelas interrogações realizadas no conjunto dos dados actuais.	40

Lista de Abreviaturas e Siglas

ASCII	American Standard Code for Information Interchange
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
CAS	Chemical Abstracts Service
CAS RN	Chemical Abstracts Service Registry Number
CSS	Cascading Style Sheets
ELBA	Extended Laidler Bond Additivity
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identity Descriptor
INCHI	International Chemical Identifier
IUPAC	International Union of Pure and Applied Chemistry
JPEG	Joint Photographic Experts Group
JS	JavaScript
MVC	Model View Controller
MySQL	My Structured Query Language
PHP	Hypertext Preprocessor
REST	Representational State Transfer
SQL	Structured Language Query
SGBD	Sistema de Gestão de Bases de Dados
SMARTS	Smiles Arbitrary Target Specification

SMILES	Simplified Molecular Input Line Entry System
SOAP	Simple Object Access Protocol
UML	Unified Modeling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USMILES	Unique Simplified Molecular Input Line Entry System

Capítulo 1

Introdução

A “químio-informática”, tradução do inglês *cheminformatics*, é uma área científica que utiliza metodologias informáticas para resolver problemas de Química normalmente associados à utilização de informação sobre estruturas moleculares. A informação química quantificável continua a crescer exponencialmente devido ao constante refinamento e optimização das tecnologias experimentais.

Uma das aplicações relevantes da “químio-informática” é servir como infra-estrutura de forma a permitir a criação, manutenção, acesso e exploração de grandes bases de dados. Esta área científica envolve a manipulação de quantidades de dados e ainda a aplicação de algoritmos, técnicas computacionais e métodos estatísticos aos dados. [1]

1.1 Motivação

O Grupo de Energética Molecular¹ do Centro de Química e Bioquímica da Faculdade de Ciências da Universidade de Lisboa, que tem como principal objecto de estudo as relações entre a estrutura, a energética, a reactividade e a função dos compostos químicos, compilou, da literatura, um conjunto de dados termoquímicos experimentais de diversos compostos químicos. Estes dados serviram de base à implementação do sistema *ThermInfo*². O *ThermInfo* é um sistema de informação para coligir e apresentar propriedades termoquímicas [2].

Recentemente o Grupo de Energética Molecular compilou mais dados de compostos químicos orgânicos, sendo que estes dados possuem novas propriedades para além das termoquímicas, ficando comprometida a sua integração no sistema *ThermInfo*, dado que a arquitectura do mesmo não foi concebida para albergar novas propriedades. Não planeando descartar estes dados, foi sugerido a criação de um novo modelo de base de dados que albergasse tanto dados para as actuais propriedades químicas do

¹ Grupo de Energética Molecular: <http://molenergetics.fc.ul.pt/>

² *ThermInfo*: <http://www.therminfo.com/>

ThermInfo, como para as novas, e que estivesse preparado para, no futuro, receber dados para outros tipos de propriedades.

A integração dos novos dados no sistema implica uma alteração na apresentação dos mesmos, assim como em algumas das funcionalidades do *ThermInfo*. Visto que o sistema foi implementado numa arquitectura que dificulta alterações profundas à base de dados, foi sugerido também realizar uma reestruturação na arquitectura do sistema de modo a facilitar futuras alterações e actualizações de um modo mais simples.

E assim com base nestes problemas foi concebida a proposta deste trabalho: a estruturação de uma base de dados para propriedades químicas de compostos orgânicos, e a sua concretização no sistema *ThermInfo*, passando assim a ser o *ThermInfo 2.0*.

1.2 Objectivos

Os principais objectivos desde projecto são criar e implementar, no sistema *ThermInfo*, um novo modelo de base de dados de compostos químicos que permita a inserção de diferentes tipos de propriedades químicas. Reestruturar a arquitectura do sistema, separando o acesso aos dados da visualização dos mesmos, de modo a que eventuais mudanças ao modelo da base de dados apenas requeiram alterações no acesso aos dados não comprometendo a sua visualização. A arquitectura do sistema será estruturada de forma a permitir, no futuro, a criação de um *Web Service*³.

Como objectivos específicos, este trabalho tem as seguintes fases:

- Familiarização com o sistema *ThermInfo*;
- Desenvolvimento e implementação do esquema relacional da base de dados;
- Importação dos dados do *ThermInfo* e dos novos dados para o novo esquema relacional;
- Alteração e implementação da arquitectura do *website* para interface com a base de dados (*ThermInfo 2.0*);
- Análise dos resultados obtidos.

1.3 Metodologia

Este projecto encontra-se dividido em 4 fases de desenvolvimento (Figura 1), interdependentes e complementares [3] [4]:

³ *Web Service* é um método de comunicação entre dois dispositivos electrónicos sobre a *world wide web*.

1. **Percepção** – É formulado e analisado o problema e são delineadas soluções com base em análises de requisitos dos utilizadores e administradores da base de dados e do sistema.
2. **Concepção** – São geradas e avaliadas as soluções, e são feitas as escolhas das soluções mais apropriadas. Para isso, são desenvolvidos diagramas de classes, utilizando *Unified Modeling Language* (UML), que modelam a base de dados e o sistema.
3. **Implementação** – É implementada a solução escolhida na fase anterior. Os dados são armazenados numa base de dados *My Structured Query Language* (MySQL⁴), com uma arquitectura adequada aos dados. A nova arquitectura do sistema *ThermInfo* é implementada em *Hypertext Preprocessor* (PHP⁵).
4. **Avaliação** – É feita uma avaliação do modelo da base de dados implementado, em termos da integridade dos dados e do tempo de resposta ao acesso aos dados. Assim como também é avaliado a estabilidade da nova arquitectura do sistema *ThermInfo 2.0*.

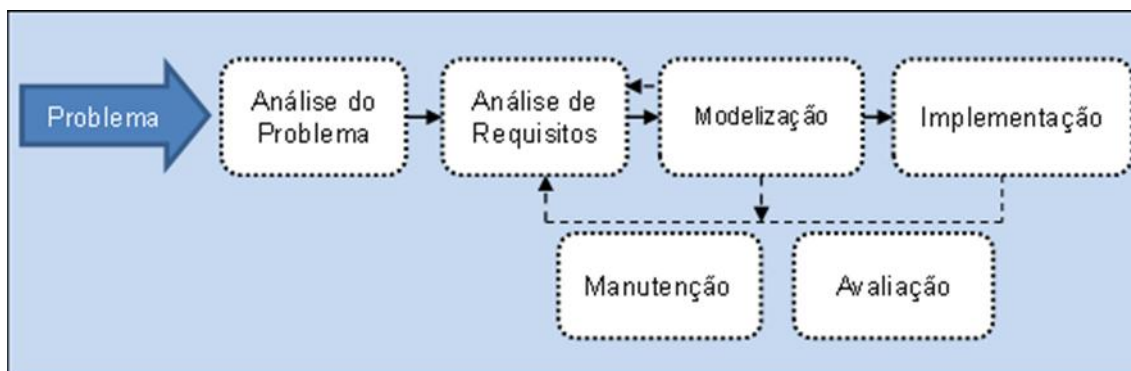


Figura 1 - Metodologia adoptada para o desenvolvimento do sistema.

⁴ MySQL: <http://www.mysql.com/>

⁵ PHP: <http://www.php.net/>

1.4 Cronograma

Na tabela seguinte (Tabela 1), é possível verificar o cronograma das actividades do projecto bem como a respectiva duração.

Tabela 1 - Cronograma das actividades do projecto (2011/2012).

Actividade/Mês	2011											2012						
	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	
Familiarização com o sistema <i>ThermInfo</i>	×																	
Familiarização com os novos dados	×																	
Pesquisa bibliográfica	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	
Preparação do ambiente de desenvolvimento	×																	
Desenvolver e implementar o esquema da base de dados		×		×														
Importar os dados para o esquema relacional					×													
Desenvolver e implementar a nova arquitectura do sistema					×	×	×	×										
Converter a interface <i>web</i> para a nova arquitectura							×	×	×			×	×	×	×	×	×	
Análise dos Resultados																×	×	
Preparar e redigir o relatório																	×	

Foram concluídas, praticamente, todas as actividades e o projecto seguiu o tempo planeado, com algumas modificações, nomeadamente a conversão da interface a análise dos resultados e a preparação do relatório foram extendidos por mais dois meses.

1.5 Organização do documento

Neste documento, e depois desta introdução que resume os objectivos e o contexto do projecto, introduzem-se no capítulo 2 alguns conceitos fundamentais à compreensão deste trabalho, assim como a descrição do sistema *ThermInfo*. No capítulo 3 descreve-se a metodologia usada no desenvolvimento do projecto. No capítulo 4 é

feita uma análise cuidada e crítica a todo o desenvolvimento realizado bem como os resultados obtidos. Finalmente no capítulo 5 são apresentadas as conclusões e direcções para trabalho futuro.

Resumidamente o relatório possui a seguinte estrutura:

1. Introdução;
2. Conceitos;
3. Metodologia de Desenvolvimento;
4. Resultados;
5. Conclusões.

Capítulo 2

Conceitos

2.1 ThermInfo

O *ThermInfo* é um sistema de informação para coligir e apresentar propriedades termoquímicas, tal como já foi referido anteriormente, contém um conjunto de dados estruturais e termoquímicos de compostos orgânicos, recolhido e avaliado pelo Grupo de Energética Molecular. O sistema encontra-se oficialmente disponível no *website* <http://www.therminfo.com/> (Figura 2) [2]. Neste capítulo vou descrever resumidamente as principais funcionalidades disponibilizadas pelo *ThermInfo*, de forma a poder, mais tarde, comparar a sua arquitectura e funcionalidades com a nova implementação. No entanto informação mais detalhada sobre o sistema pode ser encontrada em <http://www.therminfo.com/> ou em [2].

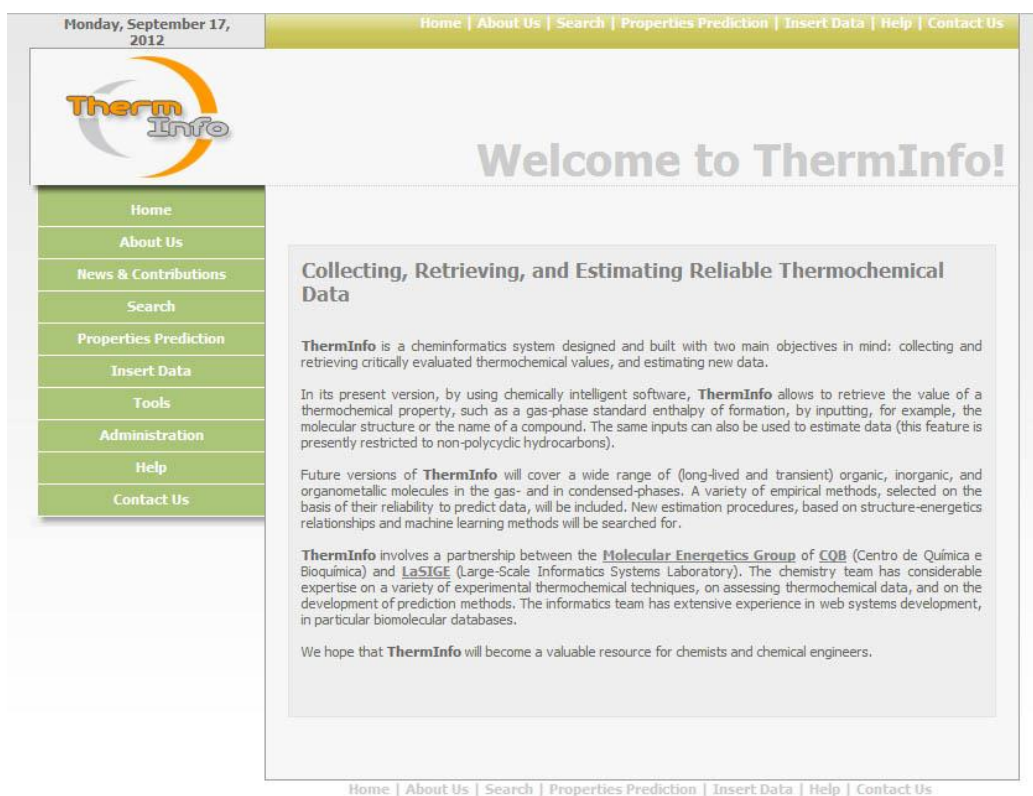


Figura 2 - Página inicial do *ThermInfo*.

Os dados contidos, actualmente, na base de dados do *ThermInfo* podem ser divididos em três categorias [2]:

1. **Dados Estruturais**, constituídos por descritores que especificam a estrutura molecular dos compostos, mostrando a forma como os átomos estão ligados, o tamanho da molécula e outras propriedades.
2. **Dados Termoquímicos**. A termoquímica estuda a energia associada a uma reacção química. A reacção é classificada como exotérmica se se realiza com libertação de energia ou endotérmica se se realiza com absorção de energia.
3. **Dados Bibliográficos**, referências completas relativamente à origem dos dados termoquímicos dos compostos.

O sistema actualmente possui uma interface que permite a realização das seguintes funcionalidades [2]:

1. **Pesquisar compostos**, a interface de pesquisa permite efectuar diferentes tipos de pesquisas:
 - Pesquisa simples – pesquisa os compostos com base num termo de pesquisa e no tipo de pesquisa seleccionado (nome, formula molecular, ID, CAS RN⁶ e SMILES⁷) (Figura 3).



Figura 3 - Interface da pesquisa simples.

- Pesquisa avançada – pesquisa os compostos com base na utilização de vários termos de pesquisa correspondentes à estrutura da molécula, restringindo assim os resultados que vão ser obtidos.

⁶ CAS Registry Number: <http://www.cas.org/content/chemical-substances/faqs>

⁷ Simplified Molecular Input Line Entry System: <http://www.daylight.com/smiles/>

- Pesquisa por estrutura – pesquisa os compostos de acordo com a sua semelhança estrutural (baseada em *fingerprints*) [5], permitindo ao utilizador desenhar a estrutura do composto, utilizando uma *applet* Java⁸, o JChemPaint⁹ (Figura 4).

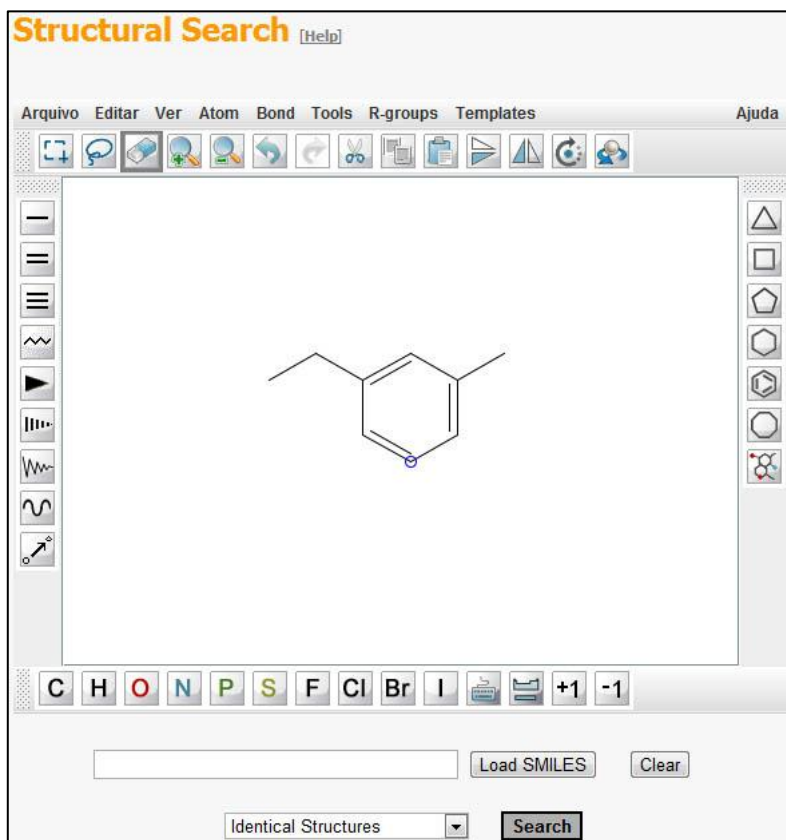


Figura 4 - Interface da pesquisa por estrutura.

- Pesquisa por subestrutura – pesquisa subestruturas de compostos que contém o fragmento estrutural pesquisado. O *input* pode ser um SMARTS¹⁰ ou o desenho da estrutura (utilizando o JChemPaint).
- Pesquisa por propriedades termoquímicas – pesquisa compostos com base em intervalos de valores de propriedades termoquímicas presentes na base de dados (Figura 5).

⁸ *Applet Java*: http://pt.wikipedia.org/wiki/Applet_Java

⁹ JChemPaint: <http://jchempaint.github.com/>

¹⁰ SMiles ARbitrary Target Specification: http://www.daylight.com/dayhtml_tutorials/languages/smarts/

Figura 5 - Interface da pesquisa por propriedades termoquímicas.

Uma pesquisa com sucesso, utilizando um destes métodos, retorna uma lista de resultados com um resumo da informação do composto, ordenada de acordo com a relevância para o(s) termo(s) da pesquisa (Figura 6). Para cada composto da lista também é disponibilizado uma ficha detalhada com toda a informação organizada de uma forma estruturada e coerente.

You are searching for: **Benzamide** ... Number of compounds found: **9**

1. Molecular ID:	CO02203
Compound Name:	Benzamide
Molecular Formula:	C ₇ H ₇ NO
CAS registry number:	55-21-0
SMILES:	O=C(N)c1ccccc1, O=C(N)C1=CC=CC=C1
More info:	View
2. Molecular ID:	CO02404
Compound Name:	Thiobenzamide
Molecular Formula:	C ₇ H ₇ NS
CAS registry number:	2227-79-4
SMILES:	NC(c1ccccc1)=S, NC(C1=CC=CC=C1)=S
More info:	View
3. Molecular ID:	CO02231
Compound Name:	2-Hydroxybenzamide
Molecular Formula:	C ₇ H ₇ NO ₂
CAS registry number:	65-45-2
SMILES:	O=C(N)c1ccccc1O, O=C(N)C1=CC=CC=C1O
More info:	View
4. Molecular ID:	CO02207
Compound Name:	1,4-Benzenedicarboxamide
Molecular Formula:	C ₈ H ₈ N ₂ O ₂

Figura 6 - Exemplo de uma lista de resultados de uma pesquisa.

- Prever propriedades termoquímicas de compostos**, permite prever apenas entalpias de formação molar padrão para hidrocarbonetos não policíclicos usando um método de aditividade - *Extended Laidler Bond Additivity* (ELBA) [6] (Figura 7).

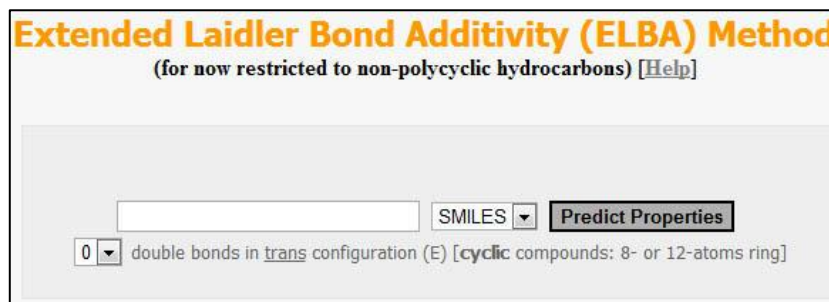


Figura 7 - Interface da previsão de propriedades termoquímicas.

3. Gerir os dados dos compostos:

- Reportar erros nos dados – disponível para todos os utilizadores, permite reportar aos administradores do sistema *ThermInfo* possíveis erros nos dados existentes na base de dados.
- Inserir dados – permite aos utilizadores registados a inserção de novos dados para compostos orgânicos no sistema *ThermInfo*.
- Apagar/Actualiza dados – permite aos administradores pesquisar por um composto com dados suspeitos ou desactualizados e proceder à sua remoção ou actualização na base de dados do sistema *ThermInfo*.
- Validar dados – permite aos administradores validar os dados relativos a compostos inseridos por utilizadores registados.
- Controlar o sistema – permite aos administradores monitorizar a evolução e o uso da base de dados do sistema *ThermInfo*. Permite também adicionar novos utilizadores que poderão inserir novos compostos orgânicos no sistema.

2.2 Representações computacionais de Estruturas

Moleculares

A base de um sistema de informação química é a sua capacidade para representar moléculas num formato legível a um computador e de transportar a estrutura de uma molécula de um lugar para outro em diferentes formatos. Há muitas maneiras para representar estruturas moleculares. Um desenho é, talvez, a mais comum e mais útil. É fácil armazenar desenhos de moléculas num computador, mas um desenho não constitui

uma representação útil. O que é necessário é uma representação que permita que as estruturas sejam armazenadas e pesquisadas. Este é, por vezes, realizada através do armazenamento de uma tabela de conexão que contém os átomos e informação de como estes se ligam numa determinada molécula. Além disso, coordenadas bi ou tridimensionais são frequentemente armazenadas. Estes dados podem ser armazenados em ficheiros ou estruturas de dados, de algumas linguagens de computador. Mas existem algumas maneiras de representar estruturas moleculares que tiram vantagens do modelo relacional de dados em um sistema de gerenciamento de base de dados relacional (SGBDR¹¹) [7].

2.2.1 SMILES

SMILES (*Simplified Molecular Input Line Entry System*) é uma representação que facilita a representação e manipulação de estruturas molecular utilizando caracteres ASCII¹². Usa os símbolos atômicos padrão para representar átomos e os símbolos: “-“ para ligações simples, “=” para ligações duplas, e “#” para ligações triplas. Átomos de hidrogénio podem ser representados explicitamente, mas são quase sempre omitidos (por exemplo, Propeno, “C=CC”; Ácido acético, “CC(=O)O”). Os SMILES contêm a mesma informação que se encontra numa tabela de conexão estendida¹³, mas com várias vantagens. É humanamente compreensível, muito compacto, e se estiver na forma canónica, representa uma cadeia única, que pode ser utilizada como um identificador universal para uma estrutura química específica.

Usando regras sobre as prioridades dos átomos, a cada estrutura pode ser atribuído um SMILES único, tendo assim um SMILES canónico. Há muitos programas de computador e aplicações de desenho de estruturas químicas que reconhecem e produzem SMILES e SMILES canónicos, assim como a conversão entre ficheiros de estruturas moleculares e SMILES (como por exemplo, Open Babel¹⁴, JChemPaint) [7] [5].

¹¹ SGBDR: http://pt.wikipedia.org/wiki/Banco_de_dados_relacional

¹² American Standard Code for Information Interchange: <http://pt.wikipedia.org/wiki/ASCII>

¹³ É uma tabela enumerando os átomos, e uma tabela enumerando as ligações e quais os átomos se encontram as ligações.

¹⁴ The Open Source Chemistry Toolbox: http://openbabel.org/wiki/Main_Page

2.2.2 SMARTS

O Uso de SMILES canónicos é uma técnica muito eficaz para o armazenamento e pesquisa de estruturas moleculares. No entanto, é por vezes necessária a realização de uma pesquisa de subestrutura, e não apenas uma pesquisa directa da estrutura. Foi assim criado a especificação SMARTS (*SMiles ARbitrary Target Specification*), uma expansão do SMILES, que permite especificar padrões moleculares e propriedades para pesquisas de subestruturas, com diferentes níveis de especificidade.

Na linguagem SMILES, existem dois tipos fundamentais de símbolos: átomos e ligações. O mesmo é verdade em SMARTS, no entanto, em SMARTS seus átomos e ligações são estendidos de modo a incluir operadores lógicos e símbolos especiais, que permitem que os átomos e ligações sejam mais gerais (por exemplo, Estrutura contendo fenol, “[OH]c1ccccc1”; Qualquer átomo com exactamente dois hidrogénios, “[*H2]”)
[5].

2.2.3 InChI

IUPAC *International Chemical Identifier* (InChI¹⁵) é um equivalente digital ao nome IUPAC para um composto. O objectivo do identificador consiste em fornecer uma sequência de caracteres capazes de forma exclusiva representar um composto químico. Uma vez que um determinado composto pode ser representado em diferentes níveis de pormenor, o identificador é representado por uma estrutura hierárquica de "camadas", onde cada camada contém uma classe distinta e separável de informação estrutural. Além da conectividade básica e carga geral, as principais variedades de camadas são: átomos de hidrogénio móveis/fixos (expressa tautomerismo), isótopos e estereoquímica (por exemplo, Etanol, “InChI=1/C2H6O/c1-2-3/h3H,2H2,1H3”).

InChI é complementado por – uma assinatura *hash*¹⁶, InChIKey. Esta assinatura é útil em aplicações de pesquisa, incluindo buscas na *web* e indexação em base de dados de estruturas moleculares (por exemplo, Etanol, “InChIKey=LFQSCWFLJHTTHZ-UHFFFAOYSA-N”).

A estrutura em camadas do identificador permite a um *software* InChI gerar diferentes identificadores para a mesma molécula, dependendo da escolha das opções

¹⁵ IUPAC International Chemical Identifier: <http://www.iupac.org/inchi/>

¹⁶ *Hash* é uma sequência de bits geradas por um algoritmo de dispersão, em geral representada em base hexadecimal, que permite a visualização em letras e números (0 a 9 e A a F).

(por exemplo, distinguir ou não tautómeros). Esta flexibilidade, no entanto, pode ser considerada uma desvantagem com relação a normalização/interoperabilidade. Para superar esta desvantagem, foi disponibilizado o InChI “padrão” (e o InChIKey padrão), que é um identificador produzido sempre com opções fixas (por exemplo, Etanol, “InChI=1S/C2H6O/c1-2-3/h3H,2H2,1H3”) [8].

2.2.4 MDL *Molfile*

Um MDL *Molfile* é um formato de arquivo, que contém informações sobre átomos, ligações, conectividade e as coordenadas de uma molécula. Este arquivo consiste num cabeçalho com informações, numa tabela de conexões que possui informações dos átomos, o tipo e a conectividade das ligações, seguido por informações mais complexas. O *Molfile* é um formato comum que muitos dos *softwares* e sistemas/aplicações de química consegue ler (Open Babel, JChemPaint). É apresentado um exemplo na Figura 8.

```
benzene
ACD/Labs0812062058

6 6 0 0 0 0 0 0 0 0 1 V2000
1.9050 -0.7932 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1.9050 -2.1232 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0.7531 -0.1282 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0.7531 -2.7882 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-0.3987 -0.7932 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-0.3987 -2.1232 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
2 1 1 0 0 0 0
3 1 2 0 0 0 0
4 2 2 0 0 0 0
5 3 1 0 0 0 0
6 4 1 0 0 0 0
6 5 2 0 0 0 0
M END
```

Figura 8 - Exemplo do conteúdo de um *molfile* do composto benzeno.

2.2.5 Fingerprints

Uma representação alternativa da estrutura molecular são as *Structural keys* (ou *Fragment Keys*). Uma *structural key* é geralmente um *array* de booleanos¹⁷ representada como um mapa de bits, ou seja, cada bit no mapa representa a presença ou ausência de uma característica estrutural específica. E podem ser usadas como filtros numa pesquisa por subestrutura. As *fingerprints* são uma família mais abstracta das

¹⁷ Uma estrutura de dados em que cada elemento é representado como verdadeiro ou falso.

structural keys, que utiliza um algoritmo para fragmentar a estrutura molecular e gravar cada fragmento da estrutura como um padrão de bits, em vez de atribuir cada fragmento a um número particular de bits como as *structural keys*.

Além de ser utilizadas nas pesquisas por subestrutura, as *fingerprints* e as *structural keys* podem ser usados como medidas de similaridade. Os padrões de bits de duas estruturas moleculares podem ser comparados considerando os bits que têm em comum, devido a fragmentos comuns. Um método popular, para a comparação de bits e quantificação da semelhança, é chamado de *Tanimoto*. Dada uma *fingerprint* para as estruturas A e B, a distância *Tanimoto* é a razão entre o número de bits que A e B têm em comum, com a soma do número de bits definidos para A e o número de bits definidos para B, menos o número de bits em comum. [5] [7]

Capítulo 3

Metodologia de Desenvolvimento

Neste capítulo será apresentada a metodologia utilizada no desenvolvimento do modelo da base de dados e da arquitectura do sistema e os detalhes das suas implementações. Inicia-se com o levantamento do contexto do problema, e uma análise de requisitos e das funcionalidades do sistema. É também apresentada a descrição e implementação do modelo da base de dados, tal como a nova arquitectura do sistema *ThermInfo 2.0*.

3.1 Análise do Problema

Nesta etapa, foram realizadas reuniões com os administradores dos dados e os administradores do sistema e analisada a documentação existente de forma a clarificar o problema e discutir propostas de resolução.

Tal como foi descrito anteriormente no Capítulo 1, é necessário o desenvolvimento de um novo modelo da base de dados de modo a albergar todos os dados compilados até agora pelo Grupo de Energética Molecular, e também dados futuros. Assim como a sua implementação com sucesso no sistema *ThermInfo*, para que se possa disponibilizar, visualizar e gerir adequadamente a informação.

3.2 Análise de Requisitos

Esta etapa tem como principais objectivos validar e sintetizar os resultados obtidos na fase anterior. Seguidamente, são apresentados os requisitos dos utilizadores e do sistema que se pretende desenvolver.

3.2.1 Utilizadores

A base de dados será aplicada ao sistema *ThermInfo 2.0*, pelo que os dados serão acedidos pelos utilizadores deste sistema, essencialmente pessoas com formação em Química. Assim como também poderá ser acedida directamente pelos administradores dos dados.

3.2.2 Funcionalidades

Dada a análise do problema e das propostas apresentadas, na concepção do modelo da base de dados, tem de se ter em conta o seguinte:

- Os dados devem ser armazenados numa base de dados com arquitectura relacional adequada.
- O armazenamento de valores de várias propriedades químicas, e a inserção de outras propriedades, se necessário.
- O relacionamento de várias referências a cada valor de uma propriedade química.
- As operações de consulta, inserção, actualização e eliminação se possam processar de forma rápida, eficiente e sem exigir elevados recursos computacionais.
- A integridade dos dados, ou seja, garantir que estes devem ser devidamente processados e administrados para evitar incorrecções.
- A capacidade de aumentar o conjunto de dados armazenados e validação por parte dos administradores.
- A manutenção, com a realização de *backups* automáticos do conjunto de dados.
- Ser totalmente integrável com o sistema *ThermInfo*.

Na Figura 9 estão representados os actores (Administradores e o Sistema *ThermInfo*), e a sua interacção com os *Use-Cases*, ou seja, as funcionalidades que a base de dados deve permitir [9]. Através do sistema *ThermInfo* tem-se acesso à base de dados para efectuar pesquisas, inserir novos dados, registar utilizadores assim como validar os novos dados e novos utilizadores, mediante um *login* no sistema. Os administradores para além do acesso à base de dados através do *ThermInfo*, podem aceder directamente à base de dados para efectuar diversas pesquisas e a gestão dos mesmos.

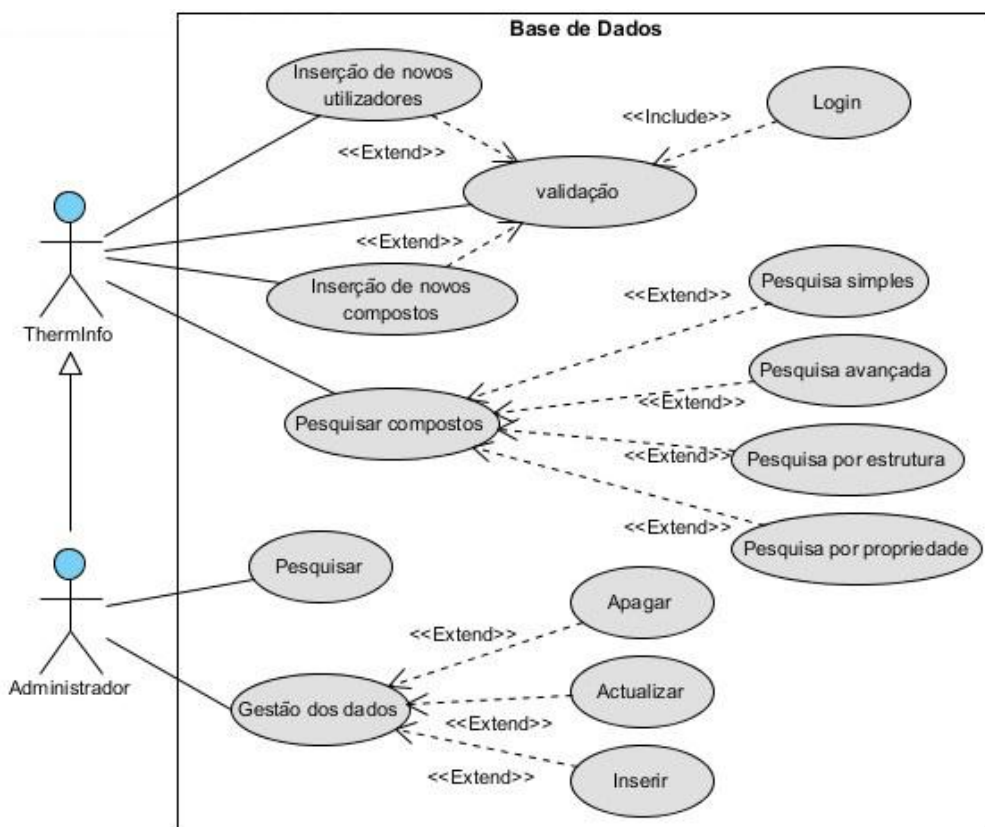


Figura 9 - Visão global das funcionalidades da base de dados em termos de actores e dependências entre *Use-cases*.

3.3 Base de Dados

Como referido anteriormente, o Grupo de Energética Molecular compilou novos dados de compostos químicos, e estes foram guardados em folhas de cálculo pelo que foi necessário juntar toda a informação para uma melhor organização (Figura 10).

A3											No. CRC					
D			E		F		G		H		I		J		K	
mation			Structural Data													
Synonym			CAS Reg. No.		Beilstein Reg. No.		Mol. Form.		Mol. Wt.		Phys. Form		SMILE		Uniqu	
			514-10-3		2221451		C ₂₂ H ₂₀ O ₂		302.451		mdl pl (al-w)					
			21293-29-8		2698956		C ₁₈ H ₁₆ O ₄		264.318		cry (chl-peth)					
5,7-Dihydroxy-2-(4-methoxyphenyl)-4H-1-benzopyran-4-one			480-44-4		277879		C ₁₈ H ₁₆ O ₅		284.263		ye nd (95% al)					
			37517-30-9				C ₁₈ H ₁₆ N ₂ O ₄		336.426		cry					
			77-46-3		2164071		C ₁₀ H ₈ N ₂ O ₂ S		332.374		pa ye nd (eth) lf (dil al)					
1,2-Dihydroacenaphthylene			83-32-9		386081		C ₁₂ H ₁₀		154.207							
Acenaphthalene			208-96-8		774092		C ₁₂ H ₈		152.192							
			82-86-0		879172		C ₁₂ H ₈ O ₂		182.175		ye nd (HOAc)					
Nicoumalone			152-72-7				C ₉ H ₈ NO ₂		353.325		cry (ace aq)					
1-[10-[3-(Dimethylamino)propyl]-10H-phenothiazin-2-yl]ethanone			61-00-7		40187		C ₁₉ H ₁₈ N ₂ OS		326.455		oran oil					
			33665-90-6				C ₁₂ H ₁₀ NO		163.153		nd (bz)					
Ethanal			75-07-0		505984		C ₂ H ₄ O		44.052		vol liq or gas					
			935-07-9				C ₃ H ₈ N ₂		134.178							
Acetaldehyde oxime			107-29-9		1209252		C ₂ H ₅ NO		59.067		nd					
Ethanamide			60-35-5		1071207		C ₂ H ₅ NO		59.067		trg mcl (al-eth)					
N-Phenylacetamide			103-84-4		606468		C ₉ H ₉ NO		135.163							
N-[5-(Aminosulfonyl)-1,3,4-thiadiazol-2-yl]acetamide			59-66-5		212994		C ₁₀ H ₈ N ₄ O ₂ S ₂		222.246							
Ethanoic acid			64-19-7		506007		C ₂ H ₄ O ₂		60.052		col liq					
			114-83-0		742880		C ₂ H ₅ N ₂ O		150.177		hex pr (eth)					
Acetyl acetate			108-24-7		385737		C ₆ H ₁₀ O ₄		102.089		liq					
			102-01-2		473419		C ₁₀ H ₁₁ NO ₂		177.2		pr or nd (bz or lig)					
			541-50-4		1747690		C ₆ H ₁₀ O ₄		102.089		cry (eth)					
2-(Methacryloyloxy)ethyl acetoacetate			21282-97-3				C ₁₄ H ₁₈ O ₆		214.215		liq					
Acetamide, 2-chloro-N-(ethoxymethyl)-N-(2-ethyl-6-methylphenyl)-			34256-82-1				C ₁₈ H ₂₅ ClNO ₂		269.768		ye liq					
			968-81-0				C ₁₈ H ₂₅ N ₂ O ₂ S		324.396		cry (EtOH aq)					
Introduction											Abbreviations		References for Critical Const		CRC Database	

Figura 10 - Conjunto dos novos dados organizados em folhas de cálculo.

Tendo em conta os dados existentes, actualmente, no *ThermInfo* e os novos dados, a informação foi toda organizada e dividida da seguinte maneira:

1. Informação geral

- Figura (em formato JPEG), representa a geometria molecular do composto, bidimensionalmente;
- ThermInfo ID, é um identificador único para cada um dos compostos, atribuído pelo sistema *ThermInfo*. Este ID tem o formato CONNNNNNN (N é um dígito);
- Nome do composto, é o nome atribuído a cada um dos compostos, baseado numa nomenclatura sistemática de acordo com as recomendações da *International Union of Pure and Applied Chemistry* (IUPAC¹⁸);
- Outros nomes, outros sinónimos de uso comum;
- Número de registo CAS (CAS RN), é um identificador único, instituído e atribuído a cada substância química pelo CAS¹⁹. Este identificador não tem qualquer significado químico e é atribuído numa ordem sequencial, de forma a assegurar a unicidade. Tem o formato NNNNNNN-NN-N (1 a 7 dígitos, hífen, 2 dígitos, hífen, 1 dígito). O último dígito é um dígito de controlo para verificar a validade e unicidade do identificador e é calculado da seguinte forma: multiplicar o último dígito por 1, o dígito seguinte por 2 e assim sucessivamente; todos estes produtos são somados; finalmente é computado o módulo 10 da soma. Por exemplo, o CASRN do metanol é 67-56-1, o dígito controlo 1 é calculado da seguinte forma: $(6*1 + 5*2 + 7*3 + 6*4) = 61$; $61 \bmod 10 = 1$;
- Número de registo Beilstein (Beilstein RN), é um identificador único de compostos na base de dados Beilstein.

¹⁸ IUPAC recommendations: <http://www.iupac.org/web/ins/2001-043-1-800>

¹⁹ Chemical Abstracts Service: <http://www.cas.org/about-cas/faqs>

2. Dados estruturais

- Fórmula molecular, dá indicação dos elementos químicos presentes e do número de átomos de cada um desses elementos [10] [11]. No sistema, os elementos encontram-se dispostos na seguinte ordem, CHXNOS (C = carbono, H = hidrogénio, X = halogéneo [flúor, cloro, bromo, iodo], N = azoto, O = oxigénio, S = enxofre);
- Peso molecular, é a soma dos pesos atómicos de todos os átomos que constituem a molécula. Indica quantas vezes uma molécula é mais pesada que a duodécima parte de um átomo de carbono-12 [10] [11];
- Estado físico, indica a situação em que o composto se encontra no que diz respeito às suas propriedades e ao movimento das partículas, dependendo da temperatura e pressão [10] [11]. Pode se dividir em:
 - Cristalino: os compostos possuem tamanho e forma definidos porque os seus átomos estão muito próximos, ligados por forças de coesão consideráveis e possuem ligeira vibração no que diz respeito à sua posição média.
 - Líquido: os compostos possuem propriedades intermédias entre os sólidos e os gases.
 - Gasoso: a principal característica dos compostos no estado gasoso é a mobilidade dos seus componentes, o que permite às substâncias ocuparem todo o volume dos recipientes que os contêm. O movimento é desordenado e as forças de interacção entre as moléculas são fracas.
- Forma física, uma notação da cor, tipo de cristais, ou outras características do composto à temperatura ambiente;
- InChI (*International Chemical Identifier*), é um identificador que consiste numa sequência de caracteres capazes de forma exclusiva representar um composto químico [8];

- InChIKey (*International Chemical Identifier Key*), é uma assinatura *hash* do InChI, facilita a pesquisa e indexação na base de dados [8];
- Std. InChI (*Standard International Chemical Identifier*), é um InChI padrão, produzido com um número de opções fixas [8];
- Std. InChIKey (*Standard International Chemical Identifier Key*), é uma assinatura *hash* do InChI padrão [8];
- Molfile, conteúdo de um ficheiro *mdl molfile* do composto;
- SMILES (*Simplified Molecular Input Line Entry System*), é uma especificação para descrever a estrutura química das moléculas usando uma curta sequência de caracteres *American Standard Code for Information Interchange* (ASCII) [5];
- SMILES único (*Unique Simplified Molecular Input Line Entry System*), é um tipo especial e único de SMILES entre todas as possibilidades válidas, para uma dada estrutura molecular;
- Classe, divide os compostos nas grandes classes estruturais (por exemplo, cadeia aberta, cíclicos, aromáticos, poli-aromáticos, etc.) [12] [13];
- Subclasse, divide os compostos pelo tipo de átomos presentes (por exemplo, CH, CHO, CHN, etc.) ou pelo tamanho dos ciclos presentes nos compostos cíclicos (anéis de 3, 4, 5, etc. átomos) [12] [13];
- Família, separa os compostos em famílias químicas de acordo com os arranjos de átomos (grupos funcionais) mais relevantes existentes na molécula [12] [13];
- Características, são *tags* atribuídas aos compostos e estão relacionadas com a presença de um determinado grupo funcional, o qual é responsável pelas propriedades químicas ou por determinadas características físicas do composto [12] [13];

- Tipo de molécula, identifica o tipo da molécula (por exemplo, orgânico, inorgânico, etc.) [10] [11].

3. Constantes Físicas

- Ponto de fusão normal, designa a temperatura a qual uma substância passa do estado sólido ao estado líquido [10] [11];
- Ponto de ebulição normal, temperatura em que a fase líquida está em equilíbrio com o vapor a uma pressão de 760 mmHg (101.325 kPa) [10] [11];
- Massa específica, é o grau de concentração de massa em determinado volume [10] [11];
- Índice de refração, é uma relação entre a velocidade da luz no vácuo²⁰ e a velocidade da luz em um determinado meio [10] [11];
- Solubilidade, é a quantidade máxima que uma substância pode dissolver-se num líquido [10] [11].

4. Constantes críticas

- Temperatura crítica, temperatura acima da qual a substância pode existir somente na forma de gás [10] [11];
- Pressão crítica, pressão de vapor na temperatura crítica [10] [11];
- Volume molar crítico, volume específico a temperatura crítica e pressão crítica [10] [11].

5. Propriedades Termodinâmicas Padrão

- Entalpias de formação molar padrão (para as fases cristalina, líquida e gasosa), é a variação de entalpia²¹ para a reacção em que um composto químico é formado a partir dos seus elementos constituintes, cada um no seu estado de referência padrão [14] [15] [16];

²⁰ É a ausência de matéria em uma certa região do espaço.

²¹ É uma função termodinâmica, especialmente útil quando se lida com processos a pressão constante, definida por $H = E + PV$, onde E é a energia, P a pressão e V o volume.

- Entropia molar padrão (para as fases cristalina, líquida e gasosa), é o grau de desordem de um sistema [16];
- Energias de Gibbs de formação molar padrão (para as fases cristalina, líquida e gasosa), é a diferença entre variação de entalpia e a temperatura vezes a variação de entropia em uma reacção em que um composto químico é formado a partir dos seus elementos constituintes, cada um no seu estado de referência padrão [16];
- Capacidade calorífica molar (para as fases cristalina, líquida e gasosa), é a proporção de calor fornecida à unidade de quantidade de uma substância para o seu consequente aumento de temperatura [16];
- Entalpias de mudança de fase molar (para as transições sólido-líquido, líquido-gás e sólido-gás), é a variação de entalpia associada aos processos físicos de transição de estado físico [14] [15] [16];
- Entalpia de fusão molar (à temperatura de fusão normal), é a quantidade de energia necessária para que um mol²² de um elemento ou substância em equilíbrio com seu líquido passe do estado sólido para o estado líquido mantida a pressão constante [14] [15] [16];
- Entalpia de vaporização molar (à temperatura de ebulição normal), é a quantidade de energia necessária para que um mol de um elemento ou de uma substância que se encontra em equilíbrio com o seu próprio vapor, a pressão de 1 atmosfera, passe completamente para o estado gasoso [14] [15] [16].

6. Dados Bibliográficos

- Descrição de cada uma das referências associadas aos dados experimentais existentes para cada composto da base de dados, incluindo: autor(es), revista científica/título do livro, ano, volume e página(s).

²² É a unidade de medida definida como a quantidade de substância de um sistema que contém as mesmas entidades elementares que há em 0,012 kg de átomos de carbono-12.

Foi desenvolvido um modelo relacional constituído por entidades que contêm os atributos que as caracterizam e pelas relações entre elas. O modelo relacional foi escolhido uma vez que a base de dados do *ThermInfo* foi implementada neste modelo, e também tem um bom desempenho e facilidade em realizar consultas aos dados (utilizando uma linguagem de alto nível, *Structured Language Query* (SQL)), é fácil de administrar, tem ampla aceitação e está muito bem documentado (o que facilita a utilização e desenvolvimento de aplicações que trabalhem sobre a base de dados).

A Figura 11 mostra o diagrama de classes (UML) da estrutura da base de dados [17]. Podemos considerar quatro categorias: os compostos, a classificação dos compostos orgânicos, as propriedades químicas e as referências bibliográficas. Ainda podemos verificar na estrutura a relação entre os compostos as propriedades e as referências.

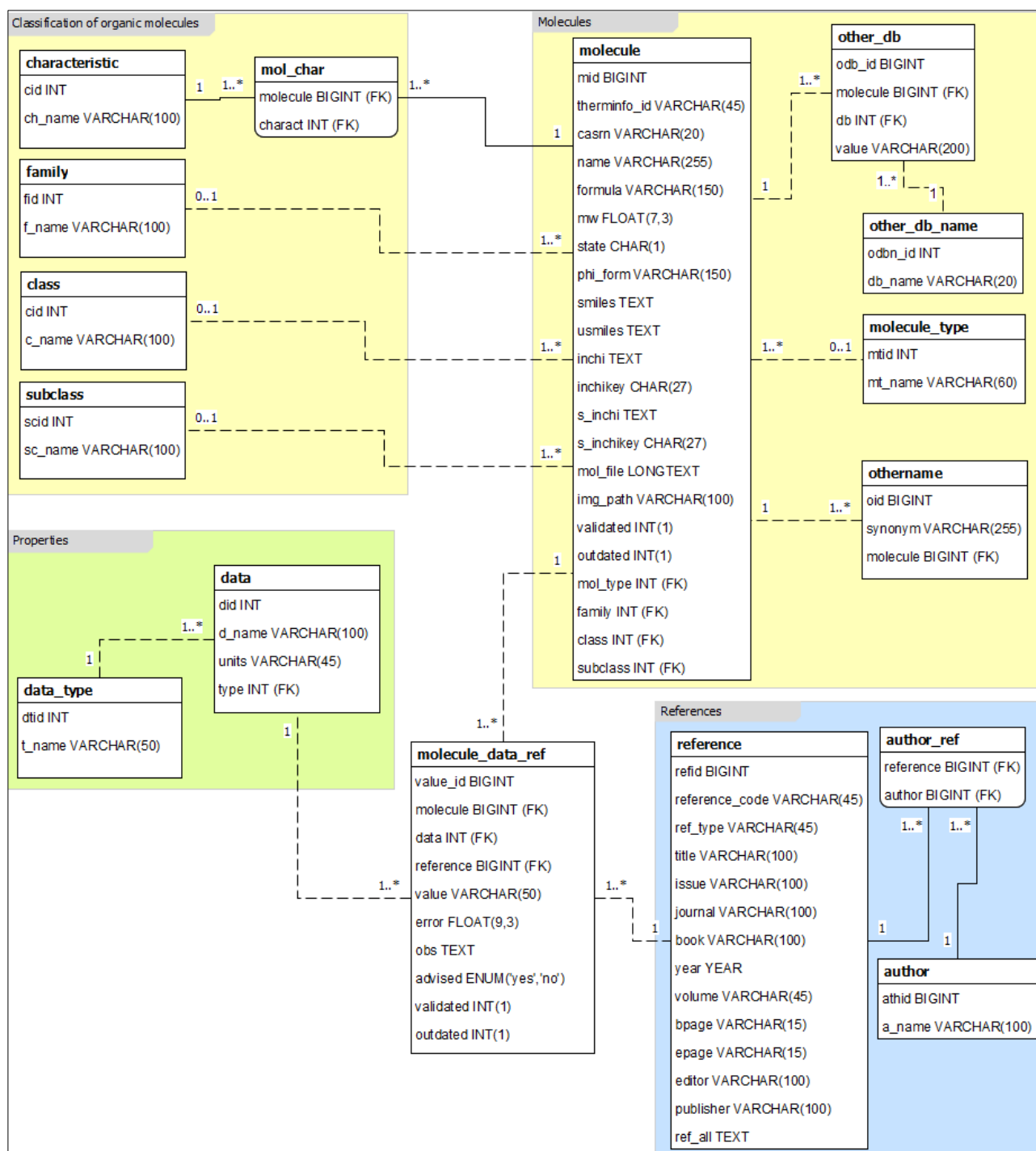


Figura 11 - Diagrama de classes UML da base de dados.

Na categoria dos compostos pode-se verificar 5 entidades, que descrevem o composto, relacionadas entre si da seguinte maneira:

- A entidade principal *molecule* contém 17 atributos (*ThermInfo* ID, CAS RN, nome, formula molecular, peso molecular, estado físico, forma física, SMILES, USMILES, InChI, InChIKey, Standard InChI, Standard

InChIKey, Molfile, imagem, e as *tags* validado e desactualizado), dos quais quer o *ThermInfo* ID quer o CAS RN são únicos. A *tag* “validado” indica se o composto foi validado ou não, assim como a *tag* “desactualizado” indica que o composto está desactualizado.

- A entidade *othername* tem um atributo que representa o sinónimo de um composto. Cada composto (*molecule*) poderá ter vários sinónimos.
- A entidade *molecule_type* tem um atributo que representa o tipo do composto (orgânico, organometálico, inorgânico, elemento). Cada composto (*molecule*) terá um tipo. Cada tipo pode ser atribuído a vários compostos (*molecule*).
- A entidade *other_db* tem como atributo uma referência a outras bases de dados, e se encontra relacionada com a entidade *other_db_name* que indica o nome da base de dados em que se fez a referência. Cada composto (*molecule*) poderá ter várias referências a outras bases de dados.

Na categoria classificação dos compostos orgânicos pode-se verificar 4 entidades, estes classificam os compostos orgânicos, e estão relacionados com a categoria dos compostos da seguinte maneira:

- As entidades *class*, *subclass* e *family* têm como atributo o respectivo nome. Cada composto (*molecule*) terá uma *class*, *subclass* e *family*. Cada *class*, *subclass* e *family* podem ser atribuídas a vários compostos (*molecule*).
- A entidade *characteristic* tem como atributo o nome da característica. Cada composto (*molecule*) poderá ter várias características, assim como cada característica pode pertencer a vários compostos (*molecule*). De forma a melhorar a performance das operações de selecção de dados nesta relação de “muitos para muitos” as colunas foram indexadas²³.

Na categoria propriedades temos 2 entidades. A entidade *data*, com 2 atributos (nome da propriedade e as unidades), que se relaciona com a entidade *data_type* que representa o tipo da propriedade (constantes físicas, constantes críticas, propriedades

²³ Índices de colunas: <http://dev.mysql.com/doc/refman/5.0/en/mysql-indexes.html>

termodinâmicas). Cada propriedade terá um tipo. Cada tipo pode ser atribuído a várias propriedades.

Na categoria referências temos também 2 entidades. A entidade *reference*, com 13 atributos (código da referência, tipo da referência, título, fascículo, jornal, livro, ano, volume, página inicial, página final, editora, publicação, referência completa), que se relaciona com a entidade *author* que representa o nome do autor da referência. Cada referência poderá ter vários autores, assim como cada autor poderá ter várias referências. De forma a melhorar a performance das operações de selecção de dados nesta relação de “muitos para muitos” as colunas também foram indexadas.

Na relação entre os compostos as propriedades e as referências estará um valor, um erro (se existir), uma observação (se for necessário), a indicação se o valor é ou não recomendado e as *tags* com indicação se o valor foi validado ou não e se está desactualizado. Esta relação é que permite que os compostos possam ter vários valores de várias propriedades, e cada um destes valores estarão associados também a várias referências, desde modo temos sempre a indicação da(s) referência(s) de cada valor. Pode-se dizer que esta relação, compostos-propriedades-referências, é o foco central da base de dados, permite ter diferentes propriedades químicas.

A base de dados foi implementada utilizando o SGBD MySQL e foi utilizado o *phpMyAdmin*²⁴ para gestão do seu conteúdo a partir de uma interface *web*. O *phpMyAdmin* permite criar, remover e alterar bases de dados e tabelas, inserir, remover e editar dados, executar interrogações SQL, exportar e importar a base de dados, entre outros processos administrativos. Estas tecnologias são as utilizadas actualmente pelo *ThermInfo*, um dos motivos da escolha das mesmas.

Primeiramente a base de dados foi carregada com dados actuais do *ThermInfo*, utilizado o PHP, foi feito a ligação à base de dados do *ThermInfo* e transportado os dados para a nova base de dados. Depois utilizando uma extensão para o PHP, o PHPExcel²⁵, foi feito a importação dos novos dados, que estavam nas folhas de cálculo, para a nova base de dados.

²⁴ *phpMyAdmin*: <http://www.phpmyadmin.net/>

²⁵ PHPExcel: <http://www.phpexcel.net/>

3.4 Arquitectura do Sistema

O *ThermInfo* foi implementado numa arquitectura *ad hoc*, ou seja, todas as funcionalidades foram agrupadas, desde as pesquisas à base de dados até a visualização da informação. Isto levou o sistema a ter uma estrutura com os seguintes componentes:

- Uma série de páginas desenvolvidas em PHP, com interrogações SQL e HTML (*HyperText Markup Language*).
- Uma série de arquivos *JavaScript* (JS) e CSS (*Cascading Style Sheets*) que estão ligadas a estas páginas.

Devido a esta estrutura alterações profundas no modelo da base de dados que suporta o sistema, implica alterações a quase todos os componentes tornando-se uma tarefa trabalhosa. Assim como o alargamento para outras funcionalidades.

Para tornar estes tipos de modificações mais fáceis, foi estudado uma maneira de separar certos componentes em partes distintas, como o acesso à base de dados, e a visualização dos dados na interface. Assim foi decidido implementar o modelo de desenvolvimento “**Model-view-controller**” (MVC²⁶).

Para o auxílio na aplicação do modelo de desenvolvimento MVC foi escolhido um *framework*²⁷ de desenvolvimento de aplicações em PHP (**CodeIgniter**²⁸).

3.4.1 Model-view-controller

O *Model-View-Controller* (MVC), como o nome indica, é um modelo de desenvolvimento de *software* que permite a separação do código em três categorias (Figura 12):

- **O modelo**, isola os dados (a lógica da aplicação);
- **A vista**, mostra os dados e elementos da interface do utilizador;
- **O controlador**, lida com os eventos do utilizador que afectam o modelo e a vista.

No caso de uma aplicação *web*, a vista seria o documento HTML gerado pela aplicação. O controlador recebe os pedidos efectuados pelo navegador *web* (GET, POST²⁹) e é gerado a vista de acordo com o modelo.

²⁶ *Model-view-controller*: <http://pt.wikipedia.org/wiki/MVC>

²⁷ **Framework** de *software* é um conjunto de classes implementadas em uma linguagem de programação específica, usadas para auxiliar o desenvolvimento de *software*.

²⁸ CodeIgniter: <http://codeigniter.com/>

Por causa da separação das partes, podem ser criados múltiplas vistas e controladores para qualquer modelo, sem a necessidade de alteração no desenho do modelo. Assim como, podem ser efectuadas alterações no modelo e ser preservada a vista. [18]

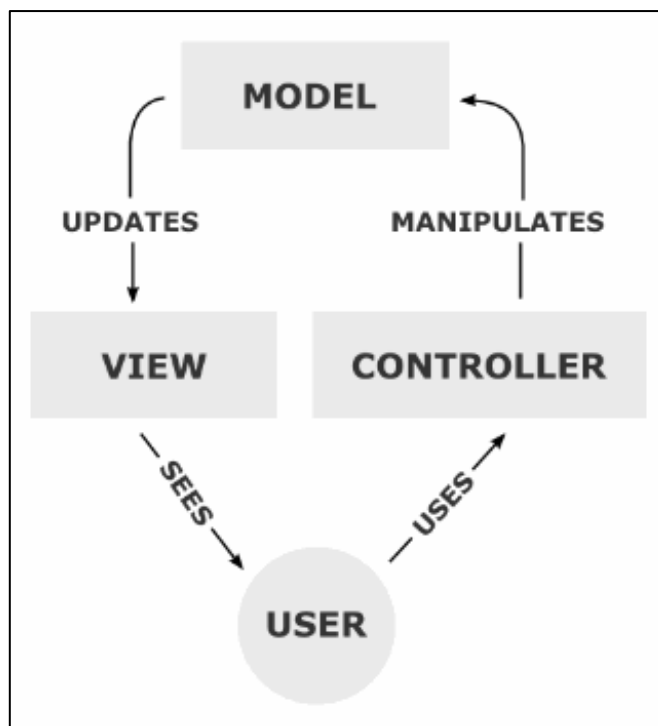


Figura 12 - Um diagrama exemplificando a relação entre o *Model*, *View* e *Controller*.

3.4.2 CodeIgniter

O CodeIgniter é um *framework* de desenvolvimento de aplicações em PHP de código aberto. Seu objectivo, por meio de um abrangente conjunto de bibliotecas voltadas às tarefas mais comuns, de uma interface e uma estrutura lógica simples para acesso àquelas bibliotecas, é possibilitar que o utilizador desenvolva projectos de *software* mais rapidamente do que se estivesse codificando do zero. O CodeIgniter usa a abordagem MVC, e permite uma separação entre a lógica e a apresentação.

A Figura 13 ilustra o funcionamento do *framework*.

²⁹ GET e POST são métodos de solicitação, do Protocolo de Transferência de Hipertexto (HTTP).

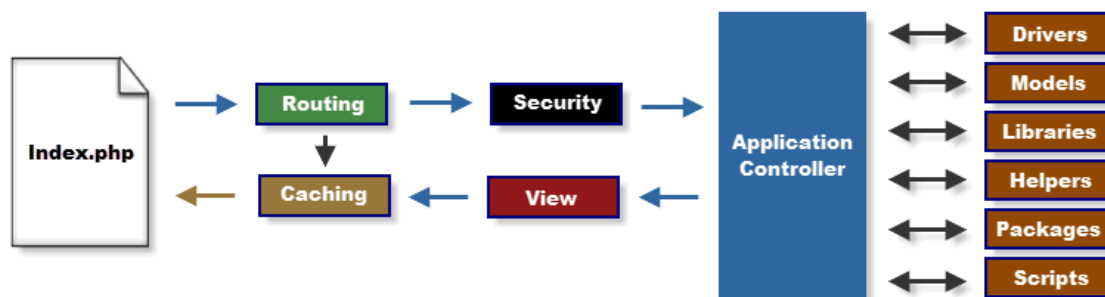


Figura 13 - Fluxograma do CodeIgniter.

1. O “*index.php*” serve como um controlador inicial, inicializa os recursos básicos necessários para executar CodeIgniter.
2. O “*routing*” examina as solicitações HTTP³⁰ para determinar o que deve ser feito com ele.
3. Se um arquivo existe em *cache*³¹, ele é enviado directamente para o navegador *web*, ignorando a execução normal do sistema.
4. Segurança. Antes do controlador da aplicação ser carregado, as solicitações HTTP e os dados enviados pelo utilizador são filtrados por segurança.
5. O controlador carrega o modelo, as bibliotecas, e quaisquer outros recursos necessários para processar a solicitação.
6. A vista é processada e então enviada para o navegador *web*. Se a *cache* estiver activada, a vista é armazenada em *cache* primeiro, para ser enviada em solicitações posteriores.

Este *framework* foi escolhido pelos seguintes motivos:

- Pelo uso da abordagem *Model-View-Controller*, necessário para a separação das partes da arquitectura do *ThermInfo 2.0*.
- Por ser leve, e isto faz com que o *ThermInfo 2.0* seja muito rápido e tenha uma boa performance.
- Pela sua flexibilidade, não segue uma convenção muito rígida podendo não ser utilizado o modelo MVC, e dá suporte a vários SGBD se algum dia for necessário uma alteração de SGBD.

³⁰ Protocolo de Transferência de Hipertexto: http://pt.wikipedia.org/wiki/Hypertext_Transfer_Protocol

³¹ Cache: <http://pt.wikipedia.org/wiki/Cache>

- Vem com uma variedade de bibliotecas e arquivos auxiliares, que ajudam nas tarefas mais comuns do desenvolvimento *web*.
- É extensível, pode ser facilmente estendido através do uso de bibliotecas e arquivos auxiliares de terceiros.
- Possui uma boa documentação.

3.4.3 *ThermInfo 2.0*

Um esquema resumo da arquitectura do sistema *ThermInfo 2.0* é apresentado na figura seguinte (Figura 14).

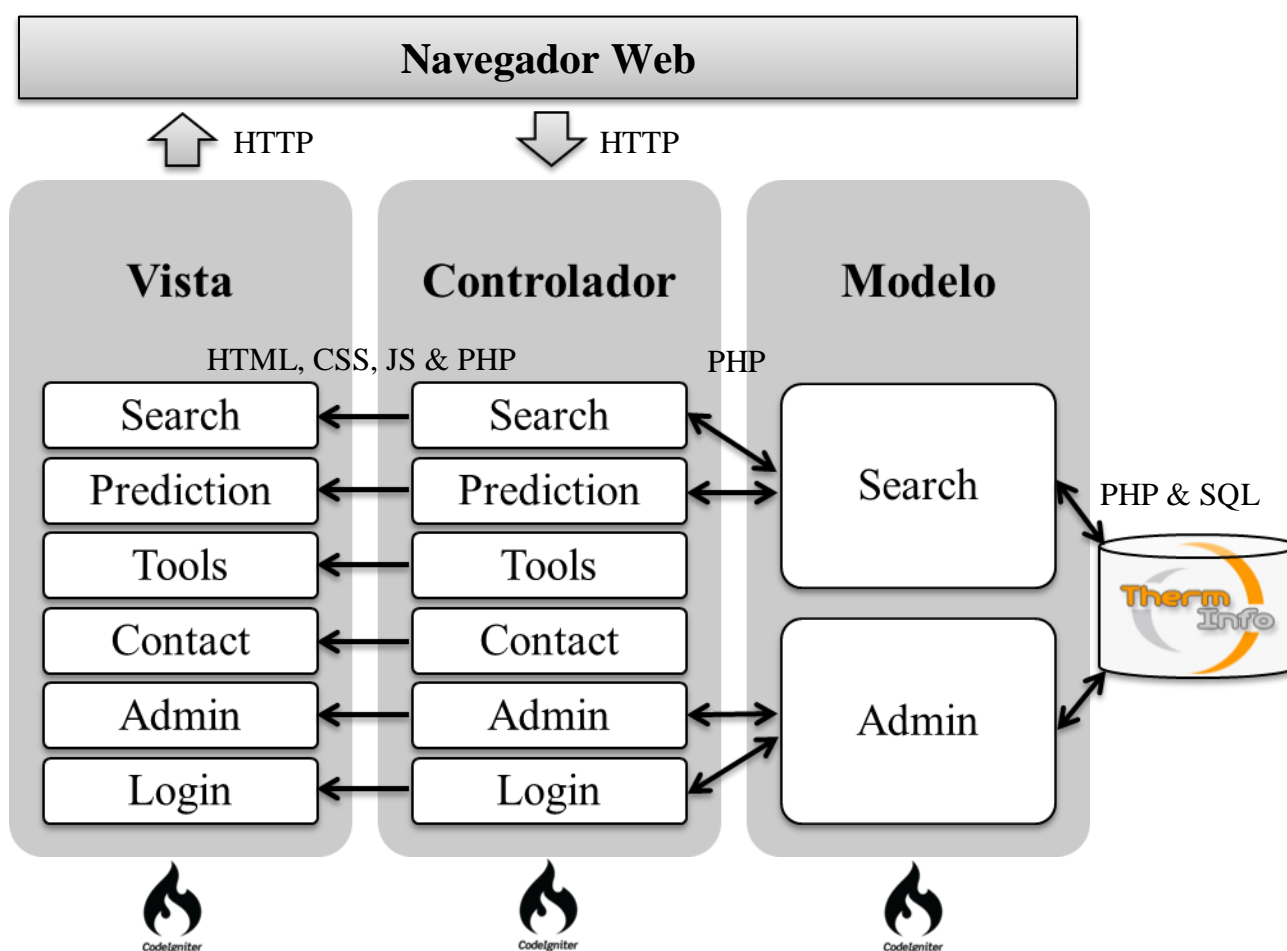


Figura 14 - Esquema da arquitectura do sistema *ThermInfo 2.0*.

Neste esquema estão representadas as vistas, os controladores e os modelos do sistema, e a forma como se relacionam entre si e a base de dados. Cada página/funcionalidade do sistema tem um controlador que é responsável por invocar o modelo e, caso necessário, apresentar uma vista (uma página em HTML, CSS e JS) de acordo com os resultados obtidos do modelo. Os controladores são acedidos pelo

utilizador através de um navegador *web* (utilizando solicitações HTTP). Todo o acesso à base de dados (pesquisas, administração dos dados, utilizadores) é efectuado pelo modelo, utilizando a linguagem SQL.

3.5 Interface

A interface do sistema *ThermInfo 2.0* tem como principais objectivos disponibilizar as funcionalidades do sistema via *web* de uma forma simples.

Foi mantido o mesmo esquema da interface do sistema *ThermInfo* actual, com pequenas alterações. A interface foi desenvolvida utilizando PHP, HTML, CSS e *JavaScript* (JS) [2]. A utilização da nova versão do HTML (HTML5), para uma melhoria na acessibilidade³² da interface, foi uma das alterações.

3.5.1 Front-end

O esquema da interface manteve-se o mesmo (Figura 15), apresenta uma barra no topo com a data o logotipo e um título, apresenta um menu lateral à esquerda com as diversas funcionalidades disponíveis, o conteúdo à direita do menu e um rodapé no fundo. O esquema irá manter-se para todas as páginas.

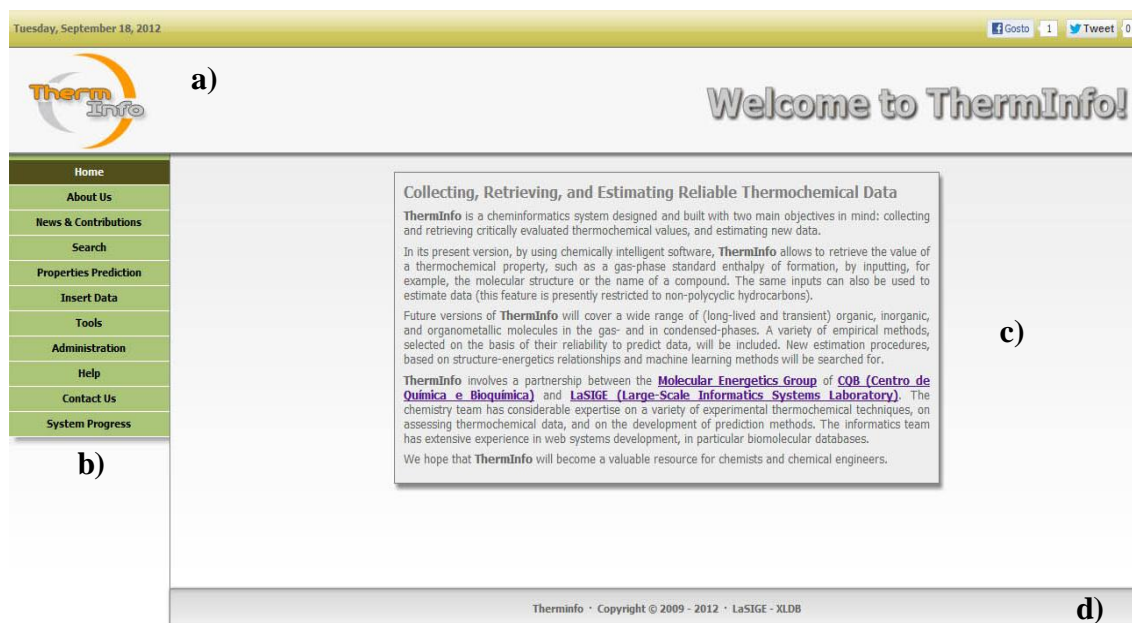


Figura 15 - Página inicial da interface do *ThermInfo 2.0*. a) Barra do topo. b) Menu lateral com as páginas/funcionalidades. c) Conteúdo. d) Rodapé.

³² Acessibilidade Web: http://pt.wikipedia.org/wiki/Acessibilidade_Web

Todos os formulários das pesquisas (pesquisa rápida, pesquisa avançada, pesquisa por estrutura e subestrutura e pesquisa por propriedades) foram alterados com as novas etiquetas do HTML5. Manteve-se nas pesquisas o CAPTCHA³³ para diferenciar humanos e máquinas (Figura 16, Figura 17 e Figura 18). O principal objectivo deste código de segurança no sistema é evitar *spam* e impedir que *software* malicioso automatizado aceda inadvertidamente à base de dados.



Figura 16 - Formulário de ‘Pesquisa Simples’, na qual pode ser especificado um termo de pesquisa e o tipo de pesquisa a efectuar.

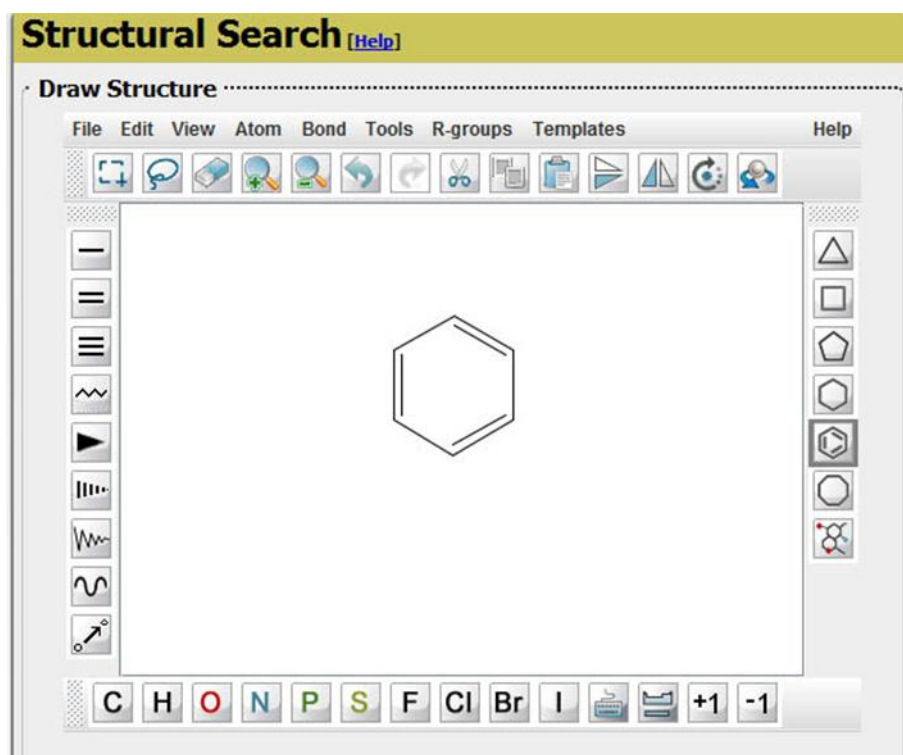


Figura 17 - Formulário de ‘Pesquisa por Estrutura’, no qual pode ser desenhado, na *applet Java*, a estrutura do composto para pesquisa.

³³ Completely Automated Public Turing Test to Tell Computers and Humans Apart: <http://captcha.net/>

Advanced Search [\[Help\]](#)

General

Compound Name:

Physical State:

Molecular Weight:

Molecular Formula:

Wildcard: ? represents one character

SMILES:

Classes and Family [\[more\]](#)

Characteristics [\[more\]](#)

Security code

[Type only **numerical characters**. Ignore letters and special characters.]

Figura 18 - Formulário de ‘Pesquisa Avançada’, no qual podem ser especificados os termos de pesquisa com mais detalhe.

Os resultados da pesquisa são apresentados ao utilizador sob a forma de listagem. Esta listagem inclui a informação relativa ao nome do composto, fórmula molecular, *ThermInfo* ID, CASRN e SMILES. São apresentados no máximo 100 compostos, contudo, o número real de compostos que satisfazem as condições da pesquisa é apresentado (Figura 19).

You are searching for: ccccccc	
... Number of compounds found: 3 ...	
- Identical structures, with score \approx 100%	
1.	
ThermInfo ID:	CO0001852
Name:	1,3,5,7-Cyclooctatetraene
Formula:	C ₈ H ₈
CAS RN:	629-20-9
SMILES:	C1=CC=CC=C1
Similarity:	100 %
More Info:	<input type="button" value="View"/>
2.	
ThermInfo ID:	CO0001873
Name:	[16]Annulene
Formula:	C ₁₆ H ₁₆
CAS RN:	3332-38-5
SMILES:	C1=C/C=CC=CC=C/C=C/C=CC=CC=C/1
Similarity:	100 %

Figura 19 - Parte da listagem de resultados obtidos de uma pesquisa exemplo.

Cada um dos compostos tem um ficha com toda a informação que pode ser visualizada clicando no botão ‘View’ da listagem de resultados (Figura 20).

Benzene

Image

Structural Data

ThermInfo ID:	CO0002404
Compound Name:	Thiobenzamide
Other Names:	1. -> Benzothiamide; 2. -> Benzothioamide; 3. -> Benzenecarbothioamide; 4. -> Phenylthioamide
CASRN:	2227-79-4
Molecular Formula:	C ₇ H ₇ NS
Molecular Weight:	137.20
Physical State:	Crystal
SMILES:	NC(c1ccccc1)=S, NC(C1=CC=CC=C1)=S
Unique SMILES:	[C]([c]1[cH][cH][cH][cH]1)([NH2])=[S], [C]([cH]1)([NH2])=[S]
InChI:	InChI=1/C7H7NS/c8-7(9)6-4-2-1-3-5-6/h1-5H,(H2,
InChIKey:	QIOZLISABUUKJY-UHFFFAOYSA-N
Standard InChI:	InChI=1S/C7H7NS/c8-7(9)6-4-2-1-3-5-6/h1-5H,(H
Standard InChIKey:	QIOZLISABUUKJY-UHFFFAOYSA-N
Mol File:	Download

Properties

Value	Uncertainty	Reference	Observations
117	n.a.	3	

Standard Molar Enthalpy of Formation at 298.15 K - Crystalline Phase (kJ/mol)

Value	Uncertainty	Reference	Observations
44.400	1.500	1	

Standard Molar Enthalpy of Formation at 298.15 K - Gas Phase (kJ/mol)

Value	Uncertainty	Reference	Observations
147.800	2.700	1	

Molar Enthalpy of Phase Change Crystal-Gas (kJ/mol)

Value	Uncertainty	Reference	Observations
103.400	2.200	1	

References

(1) - J. B. Pedley, Thermochemical Data and Structures of Organic Compounds, Vol. 1, 1994, p. 1-571.
 (3) - David R. Lide, CRC Handbook of Chemistry and Physics (CD-ROM Version 2010), 90th Ed., CRC Press/Taylor and Francis, Boca Raton, FL.

Figura 20 - Página com a ficha de um composto contendo toda a informação relativa ao dados estruturais, às propriedades assim como a(s) respectiva(s) referência(s). Os dados encontram-se agrupados em painéis móveis.

Os formulários das previsões (previsão simples e previsão por estrutura usando o método ELBA) também foram alterados para as novas etiquetas do HTML5. Manteve-se a estrutura da página com o resultado das previsões (Figura 21 e Figura 22).

Extended Laidler Bond Additivity Method (ELBA)
 (for now restricted to non-polycyclic hydrocarbons) [\[Help\]](#)

Input

Insert a search term Name

double bonds in *trans* configuration (E) [cyclic compounds: 8- or 12-atoms ring]

Predict Properties

Figura 21 - Formulário para previsão de propriedades.

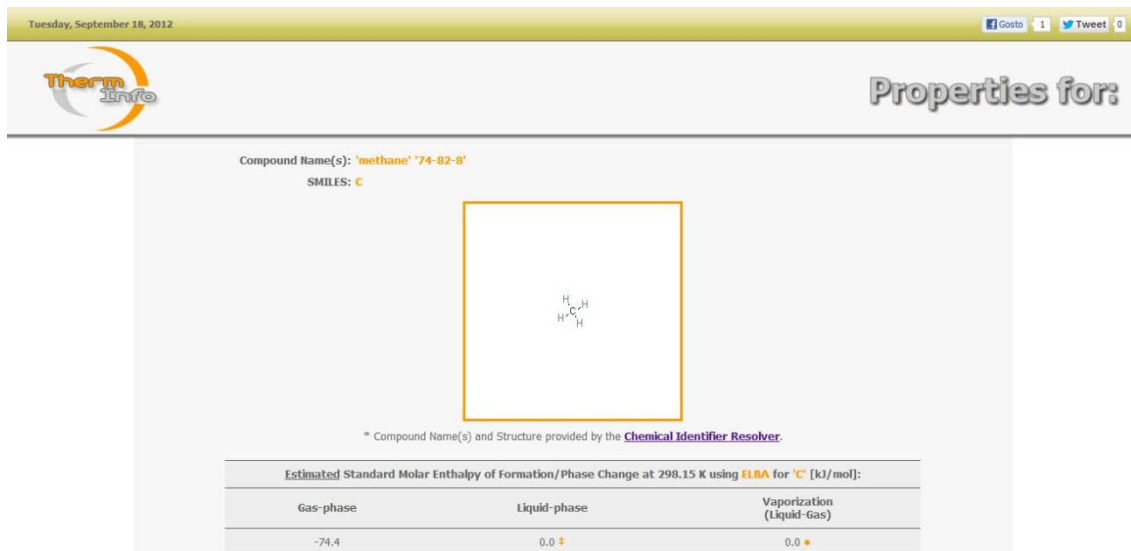


Figura 22 - Parte da página com o resultado de uma previsão de propriedades.

3.5.2 Back Office

A página da administração (*back office*) foi a que sofreu mais alterações. Mediante o *login* no sistema (Figura 23), o administrador tem acesso à página onde poderá adicionar, remover e actualizar todos os dados (compostos, classes, subclasses, características, famílias, referências, propriedades, etc.) que se encontram na base de dados. Também pode validar compostos inseridos pelos utilizadores, assim como pode gerir os utilizadores que se registam no sistema (Figura 24, Figura 25).

Através desta página, os administradores, tem o controlo da base de dados, e acesso a algumas informações sobre a utilização do sistema e evolução da base de dados.

Log in to ThermInfo! [\[Help\]](#)

User Credentials

Your e-mail address and your password are required. Note that password is case sensitive.

[You do not have an account? Register Here.](#)
[Change Your Password Here.](#)
[Forgot your password? Click Here](#)

E-mail address:

Password:

Security code

[Type only numerical characters. Ignore letters and special characters.]

Figura 23 - Formulário que permite efectuar *login* no sistema.

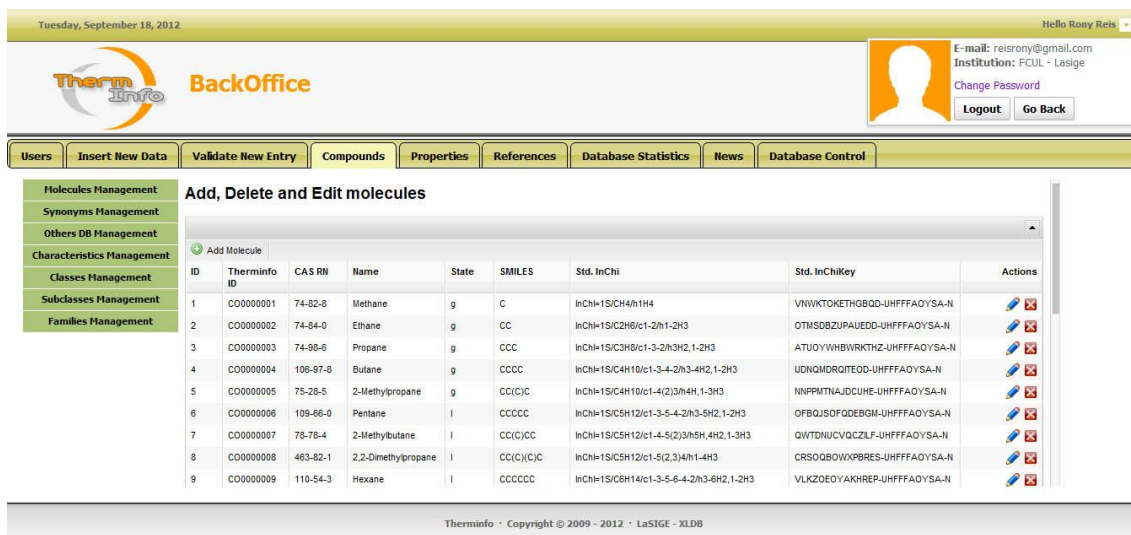


Figura 24 - Página do *back office*, onde os administradores podem gerir a base de dados.

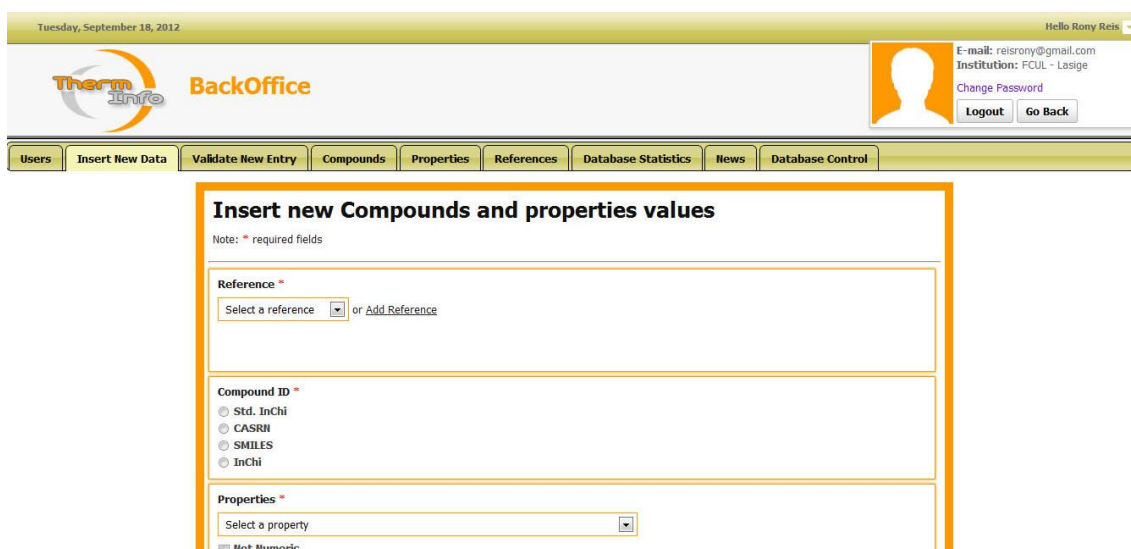


Figura 25 - Inserção de compostos no *back office*.

3.6 Web Service

O termo *Web Service* descreve uma maneira padronizada de integração de aplicações baseados na *web*, permite às aplicações enviar e receber dados em formato XML (*Extensible Markup Language*³⁴) utilizando o protocolo HTTP. Os *Web Services* são identificados por um URI (*Uniform Resource Identifier*³⁵), descritos e definidos

³⁴ XML: <http://pt.wikipedia.org/wiki/XML>

³⁵ URI: <http://pt.wikipedia.org/wiki/URI>

usando XML. Os *Web Services* são utilizados para disponibilizar serviços interativos na *web*, podendo ser utilizados por outras aplicações usando, por exemplo, o protocolo SOAP (*Simple Object Access Protocol*³⁶) ou REST (*Representational State Transfer*³⁷).

Utilizando a nova arquitectura (MVC) do *ThermInfo* 2.0 pode-se implementar com facilidade um *Web Service* para as pesquisas na base de dados e a previsão de propriedades. Para isso seria necessário a implementação de controladores para lidar com os pedido HTTP, aceder os modelos e responder com uma vista em XML (ou outro formato padrão).

Como resultado da nova arquitectura foi implementado dois controladores (pesquisa simples e a ficha do composto) para demonstrar a construção de um *Web Service* do *ThermInfo* 2.0.

Implementado em REST, o *Web Service* possui diversos recursos para cada controlador e cada recurso é identificado com um URI e pode ser acedido utilizando requisições GET. A Figura 26 mostra como está estruturado o acesso a cada recurso.

http:// url_base / controlador / recurso / parâmetro / valor / format / valor

Parâmetro **Formato**

Figura 26 - URI de acesso ao Web Service do ThermInfo 2.0.

O controlador ‘**qsearch**’ processa a pesquisa simples e possui os seguintes recursos:

- formula, pesquisa pela fórmula do composto (por exemplo, http://url_base/qsearch/formula/query/C4H6/format/xml).
- id, pesquisa pelo ThermInfo ID do composto (por exemplo, http://url_base/qsearch/id/query/CO0002343/format/xml).
- casrn, pesquisa pelo CAS RN do composto (por exemplo, http://url_base/qsearch/casrn/query/463-82-1/format/xml).
- smiles, pesquisa pelo SMILES do composto (por exemplo, http://url_base/qsearch/smiles/query/c1ccccc1/format/xml).

³⁶ SOAP: <http://pt.wikipedia.org/wiki/SOAP>

³⁷ REST: <http://pt.wikipedia.org/wiki/REST>

O controlador ‘**compound**’ processa a pesquisa de uma ficha de um composto e possui um recurso:

- **id**, identificador na base de dados do composto (por exemplo, http://url_base/compound/id/query/2343/format/xml).

O formato da resposta aparece sempre no fim do URI e é opcional, em caso de omissão o XML é o formato por defeito. Para além do XML o *Web Service* suporta outros formatos:

- **json**, subconjunto da notação de objecto de JavaScript.
- **php**, estrutura de dados do PHP.
- **serialize**, estrutura de dados do PHP serializado.

Em caso de sucesso o *Web Service* envia na resposta o código de estado HTTP 200, e em caso de erro envia os códigos 400 (erro no parâmetro), 404 (não encontrado) ou 500 (erro no sistema).

O *Web Service* encontra-se a funcionar com o seguinte URL base: <http://therminfo.lasige.di.fc.ul.pt/alpha/api/>. Este *Web Service* ainda não se encontra completo. É necessário melhoramentos em termos de segurança e outras funcionalidades. Isto será um trabalho a ser realizado no futuro.

Capítulo 4

Resultados

4.1 Base de Dados

As estatísticas da base de dados, em Julho de 2012, podem ser visualizadas na Figura 27 e Figura 28. A sua observação permite verificar a representatividade do conjunto de dados. Presentemente existem cerca de 11290 compostos orgânicos únicos e não redundantes com a maioria dos dados descritos anteriormente disponíveis e cerca de 30998 valores de propriedades, divididos em 25 propriedades químicas.

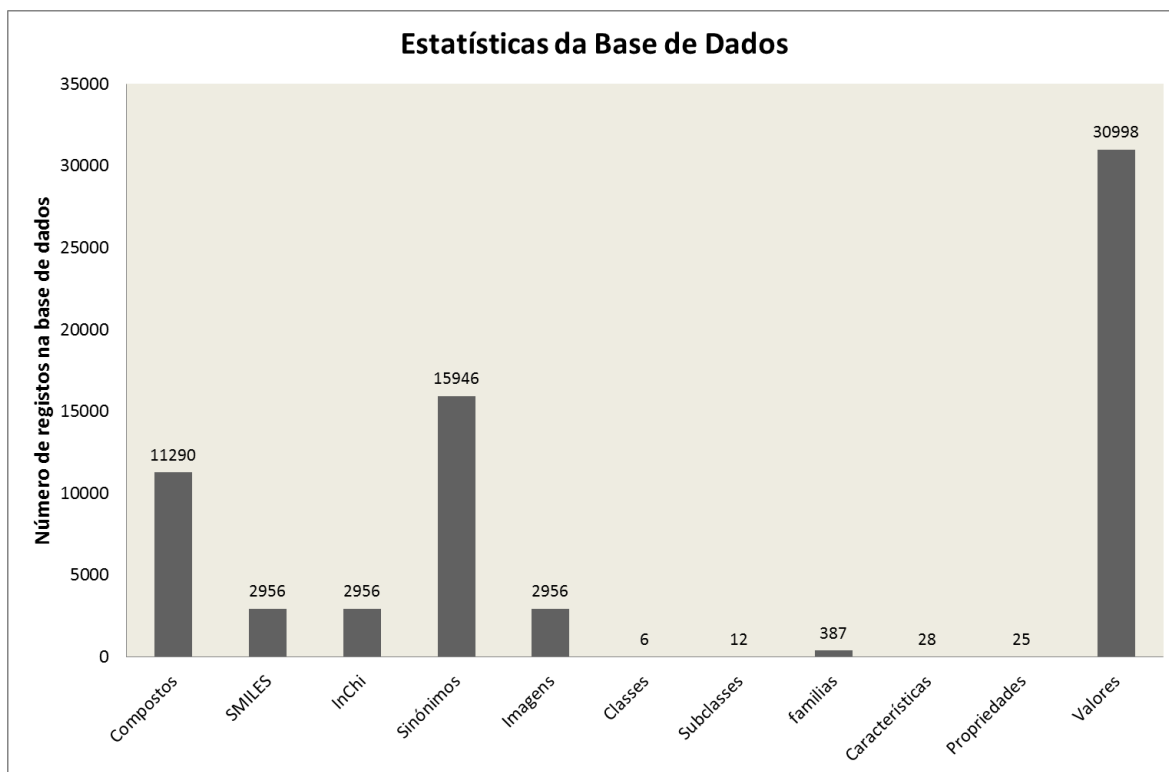


Figura 27 - Representação gráfica das estatísticas da base de dados (Registos).

Verifica-se também na Figura 28, os dados das 25 propriedades existentes na base de dados. Algumas propriedades não possuem valores até à data (5 propriedades), estes ainda não foram inseridos devido estarem incorrectos. Serão inseridos posteriormente.

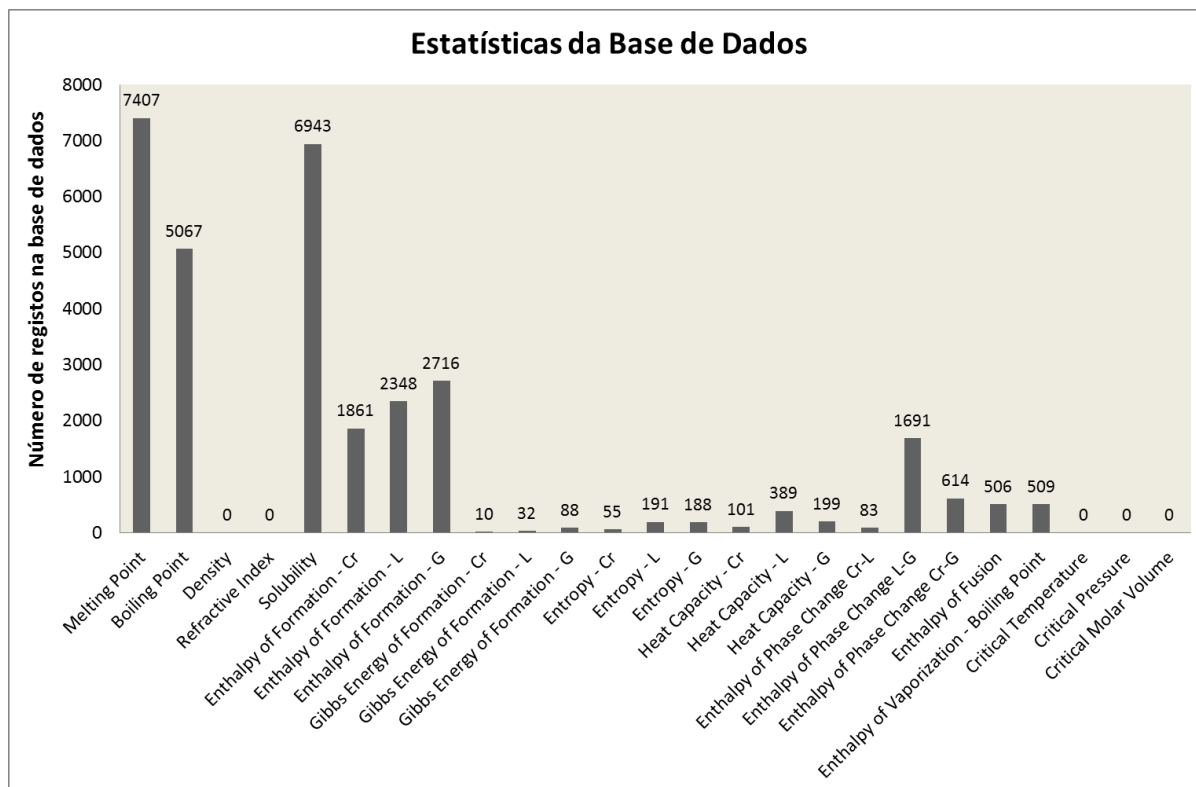


Figura 28 - Representação gráfica das estatísticas das propriedades da base de dados (Registos).

Foi efectuada uma avaliação ao desempenho das consultas/inserções realizadas sobre a base de dados para as funcionalidades disponíveis no sistema *ThermInfo 2.0*. O desempenho foi medido a partir da recolha do tempo necessário para a execução de cada interrogação (*query*) (média de 5 execuções) à base de dados e encontram-se especificados na Tabela 2.

Tabela 2 - Tempo, em segundos, utilizado pelas interrogações realizadas no conjunto dos dados actuais.

Funcionalidade	Interrogações	Tempo (segundos)	Tuplos avaliados	Tuplos retornados
Pesquisa simples por nome (por exemplo: 'methanol').	SELECT a.mid, ABS(mol.mw-32.042) AS dif FROM (SELECT mid, name, INSTR(name, 'methanol') AS fpos FROM molecule WHERE name LIKE '%methanol%' UNION SELECT molecule, synonym, INSTR(synonym, 'methanol') AS fpos FROM othername WHERE synonym LIKE '%methanol%') AS a, molecule AS mol WHERE a.mid = mol.mid AND mol.validated = 1 AND mol.outdated = 0 GROUP BY a.mid ORDER BY a.fpos, dif, a.name	0.074	27236	125

<p>Pesquisa simples por SMILES (por exemplo: 'C1=CC=CC=C1' e 100% de semelhança).</p>	<p>1) CREATE TEMPORARY TABLE IF NOT EXISTS sim_sum(mid INT NOT NULL, sim FLOAT(6,3) NOT NULL)</p> <p>2) INSERT INTO sim_sum(mid, sim) VALUES (1884, 1.000)</p> <p>3) SELECT m.mid, m.smiles, ABS(m.mw - 78.112) AS dif, sim_sum.sim AS similarity FROM molecule AS m, sim_sum WHERE m.mid IN (1884) AND sim_sum.mid = m.mid ORDER BY similarity DESC, dif ASC</p> <p>4) DROP TEMPORARY TABLE IF EXISTS sim_sum</p>	<p>0.002 + 0 + 0.001 + 0 = 0.003</p>	<p>0 + 0 + 11291 + 0 = 11291</p>	<p>0 + 0 + 1 + 0 = 1</p>
<p>Pesquisa simples por CASRN (por exemplo: '85-01-8').</p>	<p>SELECT mid FROM molecule WHERE casrn LIKE '%85-01-8%'</p>	<p>0.007</p>	<p>11290</p>	<p>1</p>
<p>Pesquisa avançada, com base em:</p> <p>- Compound: <i>propane</i></p> <p>- Molecular weight: > 50</p> <p>- Physical state: Liquid</p> <p>- Class: Ring Systems Containing Isolated Benzenoid and Non-Benzenoid Rings</p> <p>- Characteristics: Alkane, Arene</p>	<p>1) SELECT cid FROM characteristic WHERE characteristic.ch_name = 'Alkane'</p> <p>2) SELECT cid FROM characteristic WHERE characteristic.ch_name = 'Arene'</p> <p>3) SELECT molecule FROM mol_char WHERE charact IN(1, 4) GROUP BY molecule HAVING COUNT(*) >= 2</p> <p>4) SELECT m.mid, m.therminfo_id, m.casrn, m.name, m.formula, m.state, m.smiles FROM molecule AS m, class AS c, ((SELECT mid, name, INSTR(name, 'propane') AS fpos FROM molecule WHERE name LIKE '%propane%') UNION (SELECT molecule, synonym, INSTR(synonym, 'propane') AS fpos FROM othername WHERE synonym LIKE '%propane%')) AS a WHERE 1 AND m.mid IN(2531, 2532, 2533, 2534, 2535, 2536, 2584, 2585, 2586, 2587, 2840, 2841, 2842, 2843, 2844, 2845, 2846, 2847, 2848, 2850, 2883, 2884, 2886, 2887, 2888, 2889, 2890, 2892, 2893, 2895, 2896, 2897, 2947, 2948, 2949, 2950, 2951, 2952, 2953) AND a.mid = m.mid AND m.mw > '50' AND m.state = 'l' AND m.class = c.cid AND</p>	<p>0 + 0 + 0.004 + 0.08 = 0.084</p>	<p>28 + 28 + 5912 + 27236 = 33204</p>	<p>1 + 1 + 39 + 6 = 47</p>

	c.c_name = '04 - Ring Systems Containing Isolated Benzenoid and Non-Benzenoid Rings' AND m.validated = 1 AND m.outdated = 0 GROUP BY a.mid ORDER BY a.fpos, m.name			
<p>Inserir novo valor de uma propriedade para um composto existente na base de dados e uma nova referência, por exemplo:</p> <p>- Autor: ThermInfo</p> <p>- Referência: 'ThermInfo Sistema de informacao, 2012'</p> <p>- Propriedade: Temperatura crítica</p> <p>- Valor: 150</p> <p>- Composto: Butane (CCCC)</p>	<p>1) INSERT INTO author (a_name) VALUES ('ThermInfo')</p> <p>2) INSERT INTO reference (reference_code, ref_type, year, title, ref_all) VALUES ('2012/THERMINFO/BOOK', 'Book', 2012, 'ThermInfo Sistema de informacao', 'ThermInfo Sistema de informacao, 2012')</p> <p>3) INSERT INTO author_ref (reference, author) VALUES (5, 10)</p> <p>4) INSERT INTO molecule_data_ref (molecule, reference, data, value, obs, validated, advised) VALUES (4, 5, 23, 150, 'Teste', 1, 'yes')</p>	<p>0.005 + 0 + 0 + 0.011 = 0.016</p>	<p>0 + 0 + 0 + 0 + = 0</p>	<p>0 + 0 + 0 + 0 + = 0</p>

Foi efectuada uma avaliação ao desempenho da base de dados quando é enviado uma carga de interrogações por um determinado número de utilizadores (uma simulação de carga). Esta avaliação foi efectuada utilizando um *software*, que é disponibilizado junto com o SGBD MySQL para este efeito (*mysqlslap*³⁸). Para esta avaliação foi utilizada uma das pesquisas (pesquisa por nome), diferentes números de utilizadores (2, 5, 10, 20, 50 e 100) ligados ao mesmo tempo à base de dados, para cada conjunto dos utilizadores foi utilizando um conjunto diferente de total de interrogações (25, 50, 75, 100, 125 e 150). Na Figura 29, encontram-se especificados os tempos de execução para cada conjunto (utilizadores/interrogações), média de 5 testes por conjunto.

³⁸ *mysqlslap*: <http://dev.mysql.com/doc/refman/5.1/en/mysqlslap.html>

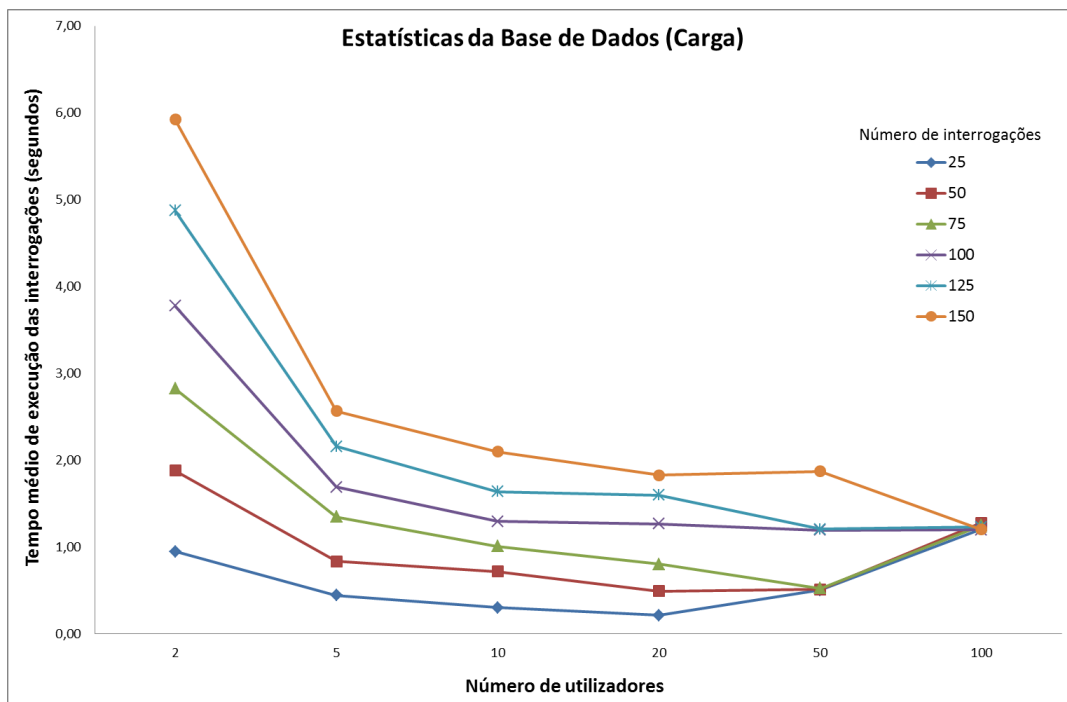


Figura 29 - Representação gráfica das estatísticas da base de dados (Carga). O tempo representa o total que leva a ser executado todas as interrogações para cada conjunto de utilizadores.

4.2 Sistema

Foi analisado a resposta do sistema em termos do tempo de execução quando se processa algumas das funcionalidades/páginas do sistema, assim como a quantidade de memória utilizada por cada uma (média de 5 requisições). Na Figura 30 encontra-se especificado o tempo total para cada execução.

Actualmente o sistema encontra-se implementado sobre um servidor com as seguintes características:

- Processador Intel® Xeon E5630 2.53Ghz;
- 8GB de memória RAM;
- Sistema operativo CentOS.

Como servidor *web* é utilizado o *software* Apache³⁹.

³⁹ Apache: <http://httpd.apache.org/>

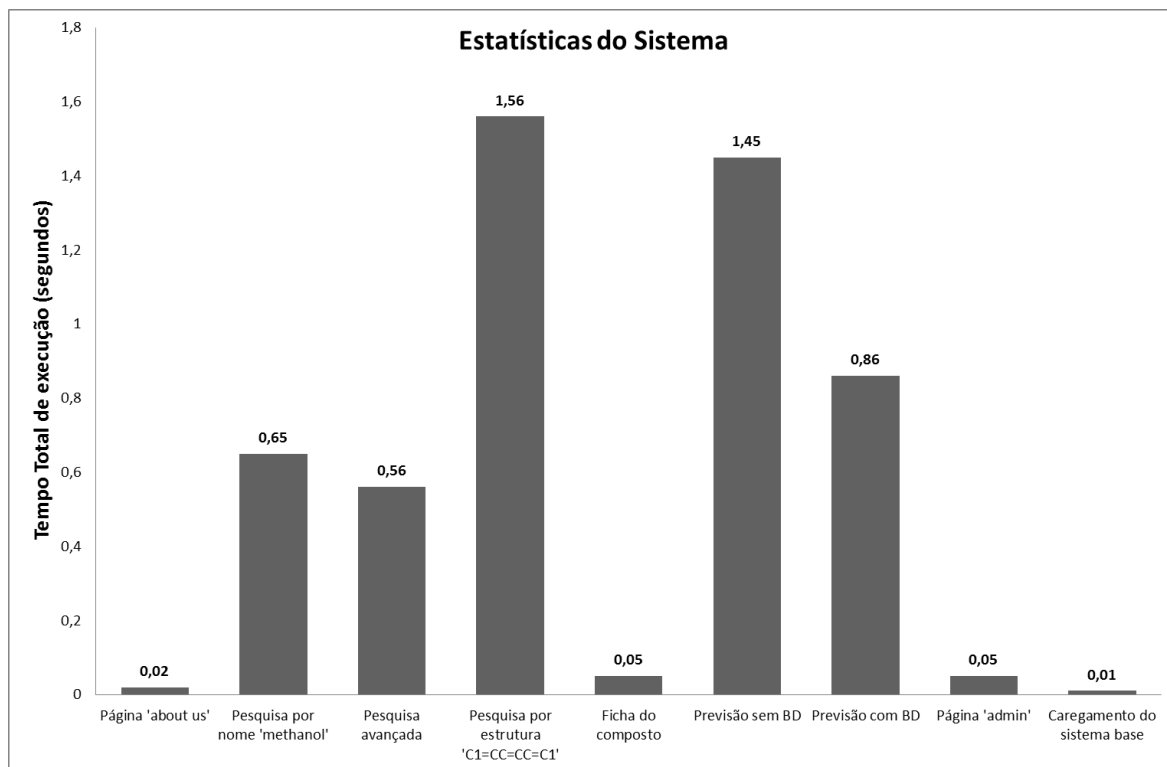


Figura 30 - Representação gráfica das estatísticas do sistema (Execução).

A Figura 31 representa a quantidade de memória que cada uma das funcionalidades/páginas que foram testadas, consomem no servidor.

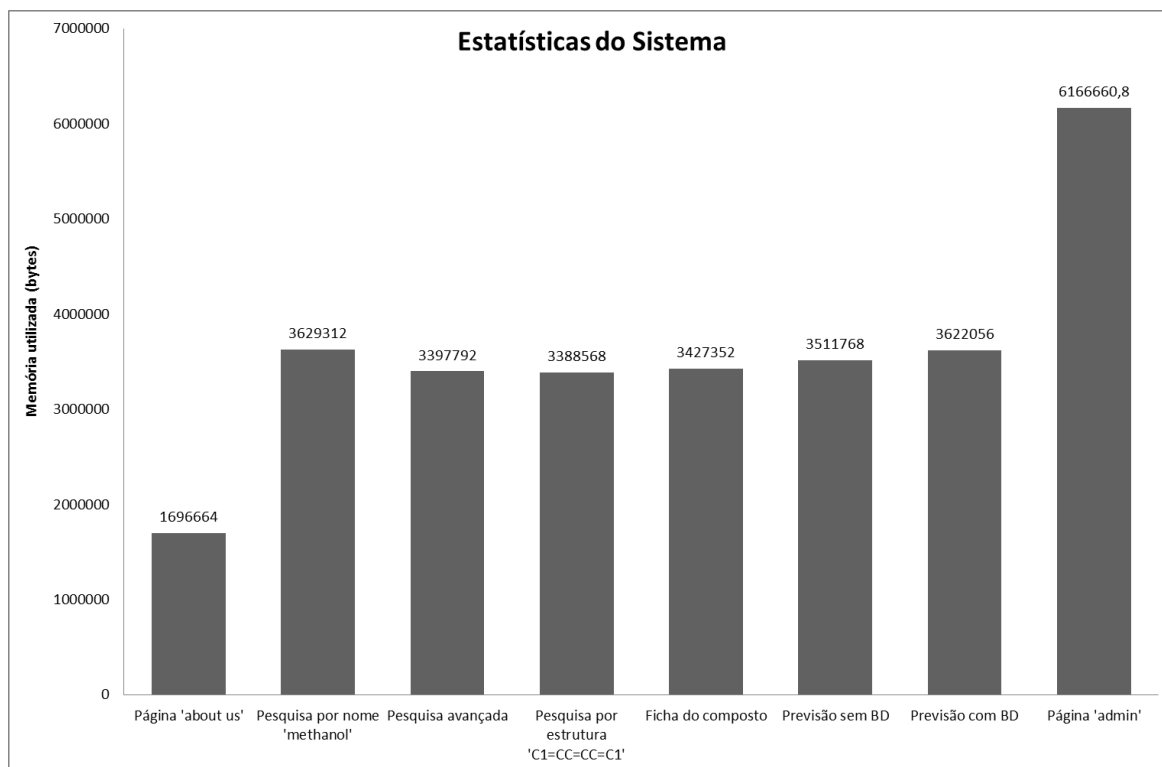


Figura 31 - Representação gráfica das estatísticas do sistema (Memória).

Referente ao desempenho do sistema foi feita uma mesma avaliação, que foi efectuada à base de dados. Foi enviado ao servidor uma carga com várias requisições de uma das páginas do sistema (página ‘about us’), por vários utilizadores ligados ao mesmo tempo. Esta avaliação foi efectuada utilizando um *software*, para este efeito, fornecido junto com o *software* do servidor *web* (*Apache HTTP server benchmarking tool*⁴⁰). Na Figura 32, encontram-se especificados os tempos médios por requisição para cada conjunto (utilizadores/requisições), média de 5 testes por conjunto.

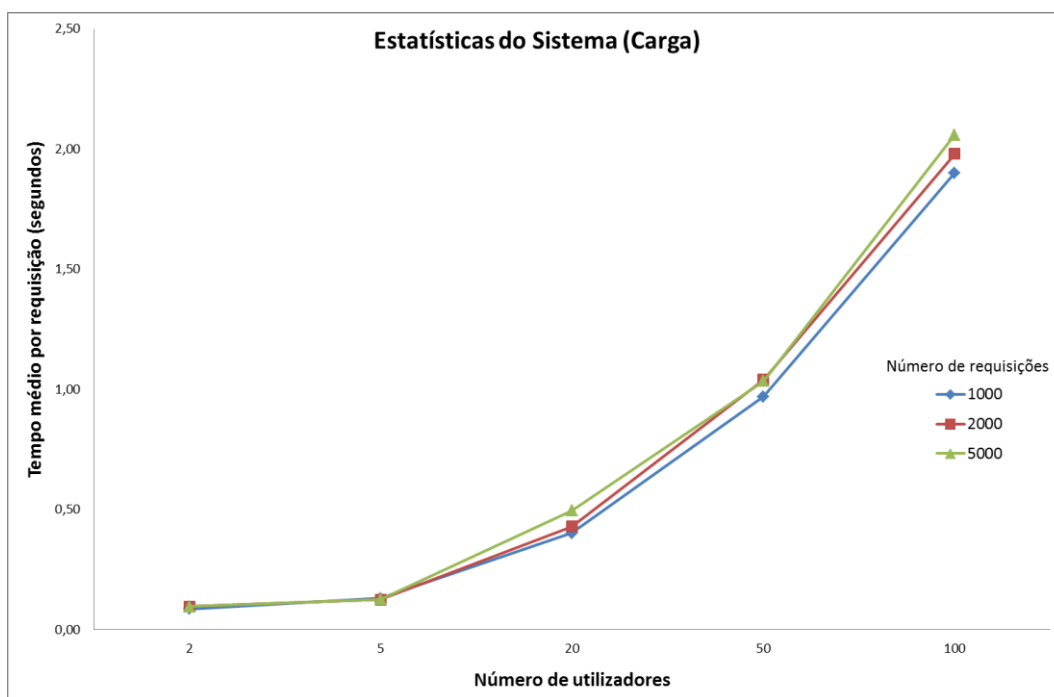


Figura 32 - Representação gráfica das estatísticas do sistema (Carga).

4.3 Análise

Analisando a Figura 27 verifica-se que do total dos 11290 compostos existentes na base de dados, apenas 2956 possui SMILES, InChI e uma estrutura química em forma de imagem, isto porque ainda não foi gerado esta informação para os restantes compostos. É necessário uma correcta identificação desta informação de modo a atribuir os compostos uma classificação precisa. Verifica-se na Figura 28 que para o total dos compostos existem cerca de 30998 valores de propriedades, divididos por 20 propriedades químicas, e de todas as propriedades existentes os pontos de ebulição e fusão e as entalpias de formação são as mais representativas.

⁴⁰ ab: <http://httpd.apache.org/docs/2.2/programs/ab.html>

Analisando os dados da Tabela 2 verifica-se que, em geral, as interrogações executadas sobre a base de dados obtiveram um tempo de resposta pequeno. O tempo de execução é prioritário para a pesquisa de dados (uma vez que o utilizador espera uma resposta imediata), tendo-se o sistema revelado ser eficiente para as pesquisas mais elaboradas (aproximadamente 0.08 segundos) e para inserção (0.01 segundos), e quase instantâneo (0.007 segundos) para as pesquisas mais simples.

Analisando a Figura 29 verifica-se o seguinte, visto que o *software* divide o conjunto das interrogações pelos utilizadores ligados, a partir de 50 interrogações por utilizador o tempo total para efectuar todas estas interrogações começa a ser maior (cerca de 4 segundos). Quando existem muitos utilizadores ligados a efectuar poucas interrogações (2 interrogações por utilizador) o tempo total de execução é pequeno (cerca de 1.5 segundos). A base de dados tem um bom desempenho quando existe muitos utilizadores ligados a efectuar poucas interrogações, desde que o número de utilizadores ligados ao mesmo tempo não ultrapasse o máximo (150 utilizadores, configuração por defeito do MySQL para um bom desempenho do SGBD com o servidor *web* Apache).

Analisando a Figura 30, verifica-se que o sistema em geral tem um tempo de resposta baixo (maior tempo aproximadamente 1.6 segundos), o que se considera ser eficiente. Verifica-se que a pesquisa por estrutura e a previsão são as funcionalidades que mais tempo leva na execução, isto porque nestas funcionalidades é utilizado o *software* Open Babel (utilizando o módulo Pybel⁴¹) para algumas conversões e cálculos para além de acessos ao *Chemical Identifier Resolver*⁴² (*web service* do grupo NCI/CADD).

Na Figura 31 pode-se verificar que em termos de utilização de memória, a que mais consome é a administração (cerca de 6MB), isto porque na página da administração existe constantes acessos aos dados da base de dados, produzindo uma maior utilização da memória do servidor. No entanto as outras funcionalidades/páginas possuem uma utilização de memória da mesma ordem (cerca de 3.5MB).

Da Figura 32 pode-se verificar que à medida que o número de utilizadores, conectados ao mesmo tempo, aumenta, o tempo de cada requisição também aumenta.

⁴¹ Pybel: http://openbabel.org/docs/current/UseTheLibrary/Python_Pybel.html

⁴² CIR: <http://cactus.nci.nih.gov/chemical/structure>

Este é um comportamento normal neste tipo de sistemas, podendo ser mitigado com o aumento de recursos do servidor.

Capítulo 5

Conclusões

Um dos principais objectivos deste projecto era a criação de uma base de dados que permitisse o armazenamento e organização de informação relativa a propriedades estruturais e químicas de compostos orgânicos e a sua respectiva bibliografia. Para isso desenvolvi uma base de dados para ser integrado numa aplicação *web* – *ThermInfo*. Um sistema de informação baseado numa base de dados para armazenar e organizar dados de propriedades termoquímicas de compostos orgânicos, com uma interface pública de fácil utilização para inserção e consulta de informação e uma interface para administração do sistema [2]. Para a integração da base de dados foi necessário também modificar a arquitectura do sistema. Foi escolhido uma arquitectura baseada no modelo MVC, para separar o sistema em várias camadas (modelo, controlador e a vista). Foi implementado com sucesso a nova versão do sistema – *ThermInfo 2.0* – encontra-se disponível na Internet através do endereço <http://therminfo.lasige.di.fc.ul.pt/alpha/main/> e está funcional para os principais navegadores *web* (Firefox, Internet Explorer, Opera, Safari e Google Chrome).

A base de dados contém, actualmente, cerca de 11290 compostos orgânicos, assim como cerca de 30998 valores de 25 propriedades químicas. A base de dados encontra-se implementada no SGBD MySQL, funcionando eficientemente e pronta para receber mais dados.

O *ThermInfo 2.0* está implementada sobre um servidor *web* Apache e possui uma interface acessível que possibilita o uso das seguintes funcionalidades por parte dos:

- Utilizadores: pesquisa simples, pesquisa avançada, pesquisa por estrutura e subestrutura, pesquisa por propriedades, previsão de propriedades e inserção de dados (mediante um registo e autenticação no sistema).
- Administradores: remoção/actualização de dados, validação de novos dados inseridos pelos utilizadores, validação do registo de utilizadores e controlo do uso do sistema.

O desenvolvimento desta base de dados e a sua implementação e integração não foi uma tarefa trivial e representou um desafio.

A concepção da base de dados não se centrou apenas na implementação, mas sim num conjunto de tarefas integrativas, nomeadamente: na modelação; na análise de requisitos a vários níveis; nos utilizadores que o vão utilizar; no tipo de funcionalidades que se pretendeu efectuar sobre a base de dados; e nos resultados da sua avaliação. A metodologia adoptada permitiu a implementação e integração da base de dados de acordo com as expectativas funcionais esperadas.

As contribuições deste projecto são o ponto de partida para a expansão de novas funcionalidades para o sistema *ThermInfo 2.0* que poderão conduzir, ainda mais, ao aperfeiçoamento do sistema.

O modelo da base de dados implementado poderá ser utilizado, com pequenas alterações, em qualquer sistema que trabalhe com moléculas e as suas propriedades. Visto que o modelo permite criar uma base de dados de compostos com vários tipos de propriedades e as suas respectivas referências.

Ao longo do desenvolvimento deste projecto, novas frentes foram abertas, que não tendo sido possível incluir neste trabalho poderão constituir direcções futuras que irão ampliar as capacidades do base de dados e do sistema *ThermInfo 2.0*. Destacam-se como exemplos:

- Estender a compilação de dados de compostos orgânicos a inorgânicos, organometálicos e radicais;
- Implementar uma classificação para compostos inorgânicos, organometálicos e radicais;
- A incorporação de outros métodos de pesquisa no *ThermInfo 2.0*, tal como por InChI e InChIKey;
- A implementação de um *web service* no *ThermInfo 2.0*;
- A implementação de novas vistas do sistema para dispositivos móveis.

Bibliografia

- [1] J. Aires de Sousa, “Químio-informática. Conteúdos que Urge Ensinar,” *BSP Química*, 84, pp. 55-59, **Janeiro 2002**.
- [2] A. Teixeira, “ThermInfo: Sistema de Informação para Coligir e Apresentar Propriedades Termoquímicas,” **2009** (Tese de Mestrado).
- [3] J. Modha, A. Gwinnett e M. Bruce, “A Review of Information Systems Development Methodology (ISDM) Selection Techniques,” *Omega*, vol. 18, n.º 5, p. 473–490, **1990**.
- [4] B. S. Blanchard e W. J. Fabrycky, *Systems engineering and analysis*, 4ª ed., New Jersey: Prentice Hall, **2006**.
- [5] Daylight Chemical Information Systems, Inc., *Daylight Theory Manual - Version 4.9*, Aliso Viejo, California: [s.n.], **2011**. Disponível na Internet em: <<http://www.daylight.com/dayhtml/doc/theory/index.pdf>> (acessado em Set. 2012).
- [6] R. C. Santos, J. P. Leal e J. Martinho Simões, “Additivity Methods for Prediction of Thermochemical Properties. The Laidler Method Revisited. 2. Hydrocarbons Including Substituted Cyclic Compounds,” *J. Chem. Thermodyn.*, 41, p. 1356–1373, **Junho 2009**.
- [7] T. O'Donnell, *Design and use of relational databases in chemistry*, Boca Raton, Flórida: CRC Press, **2009**.
- [8] S. E. Stein, S. R. Heller, D. V. Tchekhovskoi e I. V. Pletnev, *IUPAC International Chemical Identifier (InChI) - Technical Manual*, Maryland ; Moscow: [s.n.], **2010**. Disponível na Internet em: <<http://www.iupac.org/home/publications/e-resources/inchi/download.html>> (acessado em Set. 2012).
- [9] J. Rumbaugh, I. Jacobson e G. Booch, *The Unified Modeling Language Reference Manual*, Reading, Massachusetts: Addison Wesley, **1999**.
- [10] R. Chang, *Química*, 8ª ed., Lisboa: McGraw Hill, **2005**.
- [11] J. Daintith, *Facts on File Dictionary of Inorganic Chemistry*, New York: Facts On File, Inc, **2004**.

- [12] R. T. Morrison e R. Boyd, Química Orgânica, 13^a ed., Lisboa: Fundação Calouste Gulbenkian, **1996**.
- [13] J. Daintith, Facts on File Dictionary of Organic Chemistry, New York: Facts On File, Inc, **2004**.
- [14] J. Pedley, R. D. Naylor e S. P. Kirby, Thermochemical Data of Organic Compounds, 2^a ed., London ; New York: Chapman and Hall, **1986**.
- [15] J. Pedley, Thermochemical Data and Structures of Organic Compounds (TRC Data Series), College Station, Texas: Thermodynamics Research Center, **1994**.
- [16] D. R. Lide, "CRC Handbook of Chemistry and Physics (CD-ROM Version 2010)," CRC Press/Taylor and Francis, Boca Raton, **2010**.
- [17] R. Ramakrishnan e J. Gehrke, Data Management Systems, 3^a ed., E. A. Jones, Ed., New York: McGraw-Hill, **2003**.
- [18] T. Myer, Professional CodeIgniter, Indianapolis, Indiana: Wiley Publishing, Inc, **2008**.
- [19] F. Li, Developing Chemical Information Systems, New Jersey: John Wiley & Sons, **2007**.
- [20] B. Bulger, J. Greenspan e D. Wall, MySQL/PHP Database Applications, 2^a ed., Indianapolis, Indiana: Wiley Publishing, Inc, **2004**.

Apêndices

A1 – Modelo completo da Base de dados do Sistema

ThermInfo 2.0

A Figura 33 representa o modelo completo da base de dados que suporta o sistema *ThermInfo 2.0*. Ao modelo da base de dados foi adicionado uma entidade para o registro de utilizadores (*users*) do sistema e duas entidades para efeitos de estatísticas da utilização do sistema (*contador* e *dbevolution*).

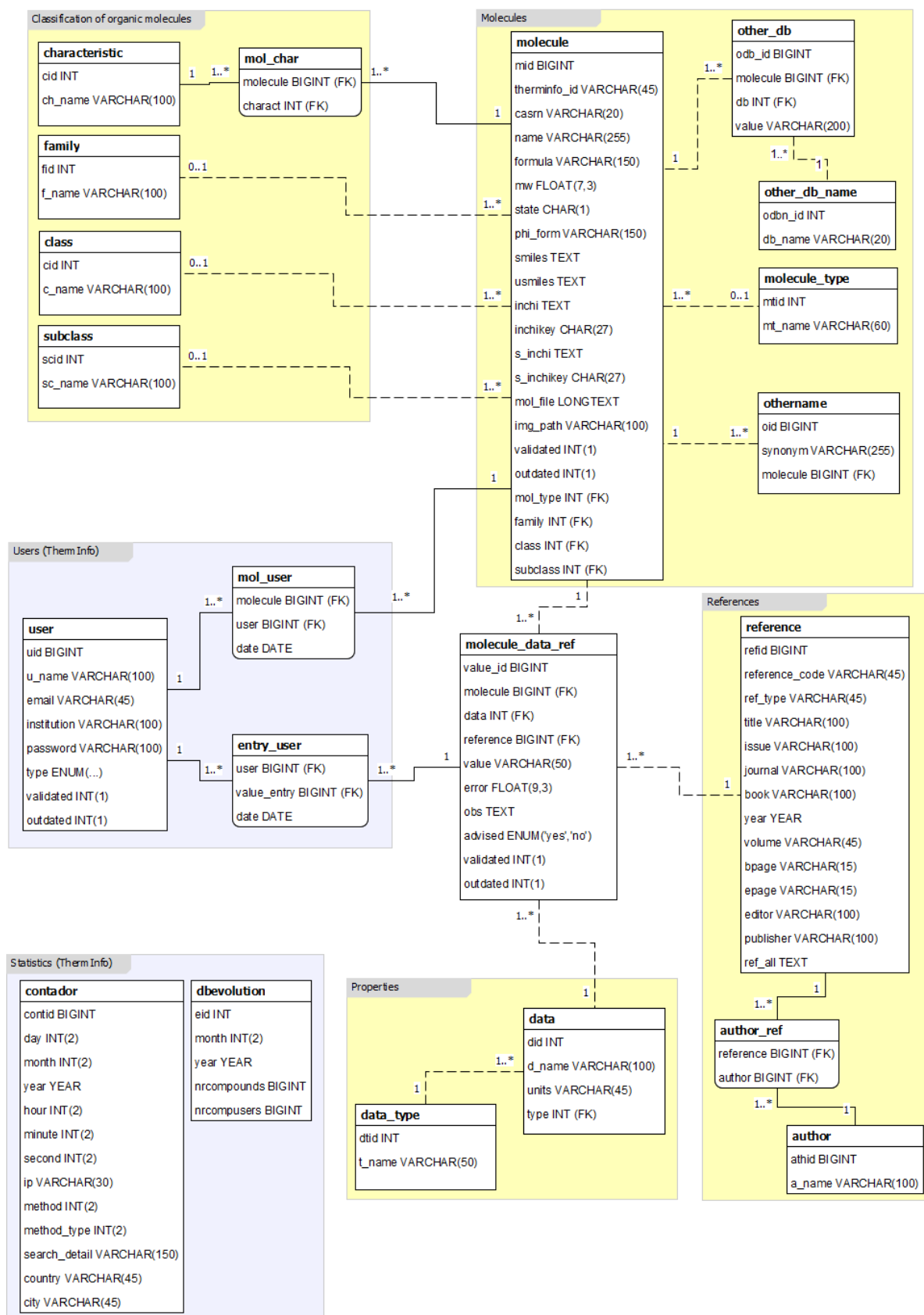


Figura 33 - Modelo da base de dados do ThermInfo 2.0.

A2 – Código SQL para implementação da Base de dados em MySQL

```
--
-- Database: therminfo2
--

DROP SCHEMA IF EXISTS therminfo2;
CREATE SCHEMA IF NOT EXISTS therminfo2 DEFAULT CHARACTER SET utf8;
USE therminfo2;

-- -----
-- (1) Table therminfo2.user
-- -----

CREATE TABLE IF NOT EXISTS user (
  uid BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
  u_name VARCHAR(100) NOT NULL,
  email VARCHAR(45) NOT NULL,
  institution VARCHAR(100) NULL,
  password VARCHAR(100) NOT NULL,
  type ENUM('guest', 'admin', 'superadmin') NOT NULL,
  validated INT(1) UNSIGNED NOT NULL DEFAULT 0,
  outdated INT(1) UNSIGNED NOT NULL DEFAULT 0,
  PRIMARY KEY (uid))
ENGINE = InnoDB;
CREATE UNIQUE INDEX ix_email ON user (email ASC);
ALTER TABLE user AUTO_INCREMENT = 1;

-- -----
-- (2) Table therminfo2.family
-- -----

CREATE TABLE IF NOT EXISTS family (
  fid INT UNSIGNED NOT NULL AUTO_INCREMENT,
  f_name VARCHAR(100) NOT NULL,
  PRIMARY KEY (fid))
ENGINE = InnoDB;
ALTER TABLE family AUTO_INCREMENT = 1;

-- -----
-- (3) Table therminfo2.class
-- -----

CREATE TABLE IF NOT EXISTS class (
  cid INT UNSIGNED NOT NULL AUTO_INCREMENT,
```



```
c_name VARCHAR(100) NOT NULL,
PRIMARY KEY (cid))
ENGINE = InnoDB;
ALTER TABLE class AUTO_INCREMENT = 1;

-- -----
-- (4) Table therminfo2.subclass
-- -----

CREATE TABLE IF NOT EXISTS subclass (
  scid INT UNSIGNED NOT NULL AUTO_INCREMENT,
  sc_name VARCHAR(100) NOT NULL,
  PRIMARY KEY (scid))
ENGINE = InnoDB;
ALTER TABLE subclass AUTO_INCREMENT = 1;

-- -----
-- (5) Table therminfo2.characteristic
-- -----

CREATE TABLE IF NOT EXISTS characteristic (
  cid INT UNSIGNED NOT NULL AUTO_INCREMENT,
  ch_name VARCHAR(100) NOT NULL,
  PRIMARY KEY (cid))
ENGINE = InnoDB;
ALTER TABLE characteristic AUTO_INCREMENT = 1;

-- -----
-- (6) Table therminfo2.molecule_type
-- -----

CREATE TABLE IF NOT EXISTS molecule_type (
  mtid INT UNSIGNED NOT NULL AUTO_INCREMENT,
  mt_name VARCHAR(60) NOT NULL,
  PRIMARY KEY (mtid))
ENGINE = InnoDB;
ALTER TABLE molecule_type AUTO_INCREMENT = 1;

-- -----
-- (7) Table therminfo2.molecule
-- -----

CREATE TABLE IF NOT EXISTS molecule (
  mid BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
  therminfo_id VARCHAR(45) NOT NULL,
  casrn VARCHAR(20) NULL,
  name VARCHAR(255) NULL,
  formula VARCHAR(150) NULL,
  mw FLOAT(7,3) NULL,
```

```

state CHAR(1) NULL,
phi_form VARCHAR(150) NULL,
smiles TEXT NULL,
usmiles TEXT NULL,
inchi TEXT NULL,
inchikey CHAR(27) NULL,
s_inchi TEXT NULL,
s_inchikey CHAR(27) NULL,
mol_file LONGTEXT NULL,
family INT UNSIGNED NULL,
class INT UNSIGNED NULL,
subclass INT UNSIGNED NULL,
mol_type INT UNSIGNED NULL,
img_path VARCHAR(100) NULL,
validated INT(1) UNSIGNED NOT NULL DEFAULT 0,
outdated INT(1) UNSIGNED NOT NULL DEFAULT 0,
PRIMARY KEY (mid),
CONSTRAINT fk_molecule_family
    FOREIGN KEY (family)
    REFERENCES family (fid),
CONSTRAINT fk_molecule_class
    FOREIGN KEY (class)
    REFERENCES class (cid),
CONSTRAINT fk_molecule_subclass
    FOREIGN KEY (subclass)
    REFERENCES subclass (scid),
CONSTRAINT fk_molecule_type
    FOREIGN KEY (mol_type)
    REFERENCES molecule_type (mtid))
ENGINE = InnoDB;

CREATE UNIQUE INDEX ix_therminfo ON molecule (therminfo_id ASC);
CREATE UNIQUE INDEX ix_casrn ON molecule (casrn ASC);
CREATE INDEX ix_inchikey ON molecule (inchikey ASC);
CREATE INDEX ix_s_inchikey ON molecule (s_inchikey ASC);
CREATE INDEX ix_molecule_family ON molecule (family ASC);
CREATE INDEX ix_molecule_class ON molecule (class ASC);
CREATE INDEX ix_molecule_subclass ON molecule (subclass ASC);
CREATE INDEX ix_molecule_type ON molecule (mol_type ASC);
ALTER TABLE molecule AUTO_INCREMENT = 1;

-- -----
-- (8) Table therminfo2.mol_user
-- -----

CREATE TABLE IF NOT EXISTS mol_user (
    molecule BIGINT UNSIGNED NOT NULL,

```

```

user BIGINT UNSIGNED NOT NULL,
date DATE NULL,
PRIMARY KEY (molecule, user),
CONSTRAINT fk_moluser_molecule
    FOREIGN KEY (molecule)
    REFERENCES molecule (mid)
    ON DELETE CASCADE,
CONSTRAINT fk_moluser_user
    FOREIGN KEY (user)
    REFERENCES user (uid)
    ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_moluser_molecule ON mol_user (molecule ASC);
CREATE INDEX ix_moluser_user ON mol_user (user ASC);

-- -----
-- (9) Table therminfo2.mol_char
-- -----

CREATE TABLE IF NOT EXISTS mol_char (
    molecule BIGINT UNSIGNED NOT NULL,
    charact INT UNSIGNED NOT NULL,
    PRIMARY KEY (molecule, charact),
    CONSTRAINT fk_molchar_molecule
        FOREIGN KEY (molecule)
        REFERENCES molecule (mid)
        ON DELETE CASCADE,
    CONSTRAINT fk_molchar_char
        FOREIGN KEY (charact)
        REFERENCES characteristic (cid)
        ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_molchar_molecule ON mol_char (molecule ASC);
CREATE INDEX ix_molchar_char ON mol_char (charact ASC);

-- -----
-- (10) Table therminfo2.othername
-- -----

CREATE TABLE IF NOT EXISTS othername (
    oid BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
    synonym VARCHAR(255) NOT NULL,
    molecule BIGINT UNSIGNED NOT NULL,
    PRIMARY KEY (oid),
    CONSTRAINT fk_othername_molecule
        FOREIGN KEY (molecule)
        REFERENCES molecule (mid)

```

```

        ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_othername_id ON othername (oid ASC);
CREATE INDEX ix_othername_synonym ON othername (synonym ASC);
CREATE INDEX ix_othername_molecule ON othername (molecule ASC);
ALTER TABLE othername AUTO_INCREMENT = 1;

-- -----
-- (11) Table therminfo2.other_db_name
-- -----

CREATE TABLE IF NOT EXISTS other_db_name (
    odbn_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
    db_name VARCHAR(20) NULL,
    PRIMARY KEY (odbn_id))
ENGINE = InnoDB;
ALTER TABLE other_db_name AUTO_INCREMENT = 1;

-- -----
-- (12) Table therminfo2.other_db
-- -----

CREATE TABLE IF NOT EXISTS other_db (
    odb_id BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
    molecule BIGINT UNSIGNED NOT NULL,
    db INT UNSIGNED NOT NULL,
    value VARCHAR(200) NULL,
    PRIMARY KEY (odb_id),
    CONSTRAINT fk_otherdb_molecule
        FOREIGN KEY (molecule)
        REFERENCES molecule (mid)
        ON DELETE CASCADE,
    CONSTRAINT fk_otherdb_name
        FOREIGN KEY (db)
        REFERENCES other_db_name (odbn_id)
        ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_otherdb_molecule ON other_db (molecule ASC);
CREATE INDEX ix_otherdb_name ON other_db (db ASC);
ALTER TABLE other_db AUTO_INCREMENT = 1;

-- -----
-- (13) Table therminfo2.data_type
-- -----

CREATE TABLE IF NOT EXISTS data_type (
    dtid INT UNSIGNED NOT NULL AUTO_INCREMENT,
    t_name VARCHAR(50) NOT NULL,

```

```

PRIMARY KEY (dtid))
ENGINE = InnoDB;
ALTER TABLE data_type AUTO_INCREMENT = 1;

-- -----
-- (14) Table therminfo2.data
-- -----

CREATE TABLE IF NOT EXISTS data (
  did INT UNSIGNED NOT NULL AUTO_INCREMENT,
  d_name VARCHAR(100) NOT NULL,
  type INT UNSIGNED NOT NULL,
  units VARCHAR(45) NULL,
  PRIMARY KEY (did),
  CONSTRAINT fk_data_datatype
    FOREIGN KEY (type)
      REFERENCES data_type (dtid))
ENGINE = InnoDB;
CREATE INDEX ix_data_datatype ON data (type ASC);
ALTER TABLE data AUTO_INCREMENT = 1;

```

```

-- -----
-- (15) Table therminfo2.reference
-- -----

CREATE TABLE IF NOT EXISTS reference (
  refid BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
  reference_code VARCHAR(45) NOT NULL,
  ref_type VARCHAR(45) NOT NULL,
  title VARCHAR(100) NOT NULL,
  journal VARCHAR(100) NULL,
  book VARCHAR(100) NULL,
  year YEAR NOT NULL,
  volume VARCHAR(45) NULL,
  issue VARCHAR(100) NULL,
  bpage VARCHAR(15) NULL,
  epage VARCHAR(15) NULL,
  editor VARCHAR(100) NULL,
  publisher VARCHAR(100) NULL,
  ref_all TEXT NULL,
  PRIMARY KEY (refid))
ENGINE = InnoDB;
ALTER TABLE reference AUTO_INCREMENT = 1;

```

```

-- -----
-- (16) Table therminfo2.author
-- -----

```

```

CREATE TABLE IF NOT EXISTS author (
  athid BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
  a_name VARCHAR(100) NOT NULL,
  PRIMARY KEY (athid))
ENGINE = InnoDB;
ALTER TABLE author AUTO_INCREMENT = 1;

-- -----
-- (17) Table therminfo2.author_ref
-- -----

CREATE TABLE IF NOT EXISTS author_ref (
  reference BIGINT UNSIGNED NOT NULL,
  author BIGINT UNSIGNED NOT NULL,
  PRIMARY KEY (reference, author),
  CONSTRAINT fk_authorref_reference
    FOREIGN KEY (reference)
      REFERENCES reference (refid)
        ON DELETE CASCADE,
  CONSTRAINT fk_authorref_author
    FOREIGN KEY (author)
      REFERENCES author (athid)
        ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_authorref_reference ON author_ref (reference ASC);
CREATE INDEX ix_authorref_author ON author_ref (author ASC);

-- -----
-- (18) Table therminfo2.molecule_data_ref
-- -----

CREATE TABLE IF NOT EXISTS molecule_data_ref (
  value_id BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
  molecule BIGINT UNSIGNED NOT NULL,
  data INT UNSIGNED NOT NULL,
  reference BIGINT UNSIGNED NOT NULL,
  value VARCHAR(50) NULL,
  error FLOAT(9,3) NULL,
  obs TEXT NULL,
  advised ENUM('yes', 'no') NOT NULL,
  validated INT(1) UNSIGNED NOT NULL DEFAULT 0,
  outdated INT(1) UNSIGNED NOT NULL DEFAULT 0,
  PRIMARY KEY (value_id),
  CONSTRAINT fk_moldataref_molecule
    FOREIGN KEY (molecule)
      REFERENCES molecule (mid)
        ON DELETE CASCADE,

```

```

CONSTRAINT fk_moldataref_data
    FOREIGN KEY (data)
    REFERENCES data (did)
    ON DELETE CASCADE,
CONSTRAINT fk_moldataref_reference
    FOREIGN KEY (reference)
    REFERENCES reference (refid)
    ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_moldataref_molecule ON molecule_data_ref (molecule ASC);
CREATE INDEX ix_moldataref_data ON molecule_data_ref (data ASC);
CREATE INDEX ix_moldataref_reference ON molecule_data_ref (reference ASC);

-- -----
-- (19) Table therminfo2.entry_user
-- -----

CREATE TABLE IF NOT EXISTS entry_user (
    user BIGINT UNSIGNED NOT NULL,
    value_entry BIGINT UNSIGNED NOT NULL,
    date DATE NULL,
    PRIMARY KEY (user, value_entry),
    CONSTRAINT fk_entry_user
        FOREIGN KEY (user)
        REFERENCES user (uid)
        ON DELETE CASCADE,
    CONSTRAINT fk_entry_value
        FOREIGN KEY (value_entry)
        REFERENCES molecule_data_ref (value_id)
        ON DELETE CASCADE)
ENGINE = InnoDB;
CREATE INDEX ix_entry_user ON entry_user (user ASC);
CREATE INDEX ix_entry_value ON entry_user (value_entry ASC);

-- -----
-- (20) Table therminfo2.contador
-- -----

CREATE TABLE IF NOT EXISTS contador (
    contid BIGINT UNSIGNED NOT NULL AUTO_INCREMENT,
    day INT(2) UNSIGNED NOT NULL,
    month INT(2) UNSIGNED NOT NULL,
    year YEAR NOT NULL,
    hour INT(2) UNSIGNED NULL,
    minute INT(2) UNSIGNED NULL,
    second INT(2) UNSIGNED NULL,
    ip VARCHAR(30) NOT NULL,

```

```
method INT(2) UNSIGNED NOT NULL,  
method_type INT(2) UNSIGNED NULL,  
search_detail VARCHAR(150) NULL,  
country VARCHAR(45) NULL,  
city VARCHAR(45) NULL,  
PRIMARY KEY (contid))  
ENGINE = InnoDB;  
ALTER TABLE contador AUTO_INCREMENT = 1;  
  
-----  
-- (21) Table therminfo2.dbevolution  
-----  
  
CREATE TABLE IF NOT EXISTS dbevolution (  
    eid INT UNSIGNED NOT NULL AUTO_INCREMENT,  
    month INT(2) UNSIGNED NOT NULL,  
    year YEAR NOT NULL,  
    nrcompounds BIGINT UNSIGNED NOT NULL,  
    nrcompusers BIGINT UNSIGNED NOT NULL,  
    PRIMARY KEY (eid))  
ENGINE = InnoDB;  
ALTER TABLE dbevolution AUTO_INCREMENT = 1;
```